



## Calhoun: The NPS Institutional Archive DSpace Repository

---

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

---

1986-03

An automated systems and project  
management system for Headquarters, U.S.  
Marine Corps

Cavallaro, Richard C.

---

<http://hdl.handle.net/10945/21690>

---

*Downloaded from NPS Archive: Calhoun*



<http://www.nps.edu/library>

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community.

Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School  
411 Dyer Road / 1 University Circle  
Monterey, California USA 93943



DUDLEY KNOX LIBRARY  
NAVAL POSTGRADUATE SCHOOL  
MONTEREY, CALIFORNIA 93943-5002





# NAVAL POSTGRADUATE SCHOOL

## Monterey, California



# THESIS

AN AUTOMATED SYSTEMS AND PROJECT MANAGEMENT SYSTEM  
FOR HEADQUARTERS, U.S. MARINE CORPS

by

Richard C. Cavallaro

March 1986

Thesis Advisor:

Norman R. Lyons

Co-advisor:

Daniel R. Dolk

Approved for public release; distribution is unlimited.

T226208



## REPORT DOCUMENTATION PAGE

|  |   |   |                             |
|--|---|---|-----------------------------|
| a. REPORT SECURITY CLASSIFICATION  |   | 1b. RESTRICTIVE MARKINGS  |                             |
| a. SECURITY CLASSIFICATION AUTHORITY   |   | 3. DISTRIBUTION / AVAILABILITY OF REPORT  |                             |
| b. DECLASSIFICATION / DOWNGRADING SCHEDULE   |   | Approved for public release;<br>distribution is unlimited   |                             |
| PERFORMING ORGANIZATION REPORT NUMBER(S)   |   | 5. MONITORING ORGANIZATION REPORT NUMBER(S)   |                             |
| a. NAME OF PERFORMING ORGANIZATION<br><br>Naval Postgraduate School  | 6b. OFFICE SYMBOL<br><i>(If applicable)</i><br>54                 | 7a. NAME OF MONITORING ORGANIZATION<br><br>Naval Postgraduate School  |                             |
| c. ADDRESS (City, State, and ZIP Code)<br><br>Monterey, CA 93943   | 7b. ADDRESS (City, State, and ZIP Code)<br><br>Monterey, CA 93943 |   |                             |
| a. NAME OF FUNDING/SPONSORING<br>ORGANIZATION  | 8b. OFFICE SYMBOL<br><i>(If applicable)</i>                       | 9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER   |                             |
| c. ADDRESS (City, State, and ZIP Code)   | 10. SOURCE OF FUNDING NUMBERS                                     |   |                             |
|  | PROGRAM<br>ELEMENT NO.  | PROJECT<br>NO.  | TASK<br>NO.                 |
|  |   |   | WORK UNIT<br>ACCESSION NO   |
| 11. TITLE <i>(Include Security Classification)</i><br>AN AUTOMATED SYSTEMS AND PROJECT MANAGEMENT SYSTEM FOR HEADQUARTERS, U.S. MARINE CORPS   |   |   |                             |
| 12. PERSONAL AUTHOR(S)<br>CAVALLARO, Richard C.  |   |   |                             |
| a. TYPE OF REPORT<br>Master's Thesis   | 13b. TIME COVERED<br>FROM _____ TO _____                          | 14. DATE OF REPORT (Year, Month, Day)<br>1986 March   | 15. PAGE COUNT<br>153       |
| 16. SUPPLEMENTARY NOTATION   |   |   |                             |
| COSATI CODES   |   | 18. SUBJECT TERMS <i>(Continue on reverse if necessary and identify by block number)</i><br>SYSTEMS MANAGEMENT, PROJECT MANAGEMENT, R:BASE DBMS |                             |
| FIELD  | GROUP   | SUB-GROUP   |                             |
|  |   |   |                             |
| 19. ABSTRACT <i>(Continue on reverse if necessary and identify by block number)</i><br>The administration of all automated information systems is a task that continues to become increasingly difficult as the numbers and complexity of these systems grow. The Department of the Navy has promulgated regulations dealing with life cycle management of these systems from the mission analysis/project initiation phase through such time that the system is terminated. The resources required to perform this managerial control are no longer adequate without the assistance of automated support. This thesis will develop a micro-oriented database management system that will enhance and simplify management control of the more than one hundred automated systems currently sponsored by the Manpower Systems Integration Branch at Headquarters, United States Marine Corps. |   |   |                             |
| 20. DISTRIBUTION / AVAILABILITY OF ABSTRACT<br><input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS  |   | 21. ABSTRACT SECURITY CLASSIFICATION<br>UNCLASSIFIED  |                             |
| 22a. NAME OF RESPONSIBLE INDIVIDUAL<br>Norman F. Lyons   |   | 22b. TELEPHONE <i>(Include Area Code)</i><br>408 646-2666   | 22c. OFFICE SYMBOL<br>541.b |

Approved for public release; distribution is unlimited.

An Automated Systems and Project  
Management System  
for Headquarters, U.S. Marine Corps

by

Richard C. Cavallaro  
Major, United States Marine Corps  
B. S., The Pennsylvania State University, 1971

Submitted in partial fulfillment of the  
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL  
March 1986

## ABSTRACT

The administration of all automated information systems is a task that continues to become increasingly difficult as the numbers and complexity of these systems grow. The Department of the Navy has promulgated regulations dealing with life cycle management of these systems from the mission analysis/project initiation phase through such time that the system is terminated. The resources required to perform this managerial control are no longer adequate without the assistance of automated support.

This thesis will develop a micro-oriented database management system that will enhance and simplify management control of the more than one hundred automated systems currently sponsored by the Manpower Systems Integration Branch at Headquarters, United States Marine Corps.

100-  
C3310  
**THESIS DISCLAIMER**

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

|      |  |    |
|------|--|----|
| I.   | INTRODUCTION . . . . .                     | 10 |
| A.   | PURPOSE . . . . .                          | 10 |
| B.   | SCOPE . . . . .                            | 11 |
| C.   | OUTLINE . . . . .                          | 11 |
| II.  | NATURE OF THE PROBLEM . . . . .            | 13 |
| A.   | OPERATIONAL ENVIRONMENT . . . . .          | 13 |
| B.   | PROJECT MANAGEMENT . . . . .               | 14 |
| 1.   | Projects for Systems Under Development . . | 14 |
| 2.   | Projects During the Maintenance Phase . .  | 14 |
| 3.   | Project Priorities . . . . .               | 16 |
| C.   | STATUS QUO . . . . .                       | 16 |
| D.   | PROPOSED SYSTEM . . . . .                  | 17 |
| III. | METHODOLOGY . . . . .                      | 19 |
| A.   | GENERAL . . . . .                          | 19 |
| 1.   | Capabilities of R:base . . . . .           | 19 |
| 2.   | Limitations of R:base . . . . .            | 20 |
| B.   | DESIGN METHODOLOGY . . . . .               | 21 |
| 1.   | Defining Requirements . . . . .            | 21 |
| 2.   | Prototyping . . . . .                      | 22 |
| IV.  | MCASMIS DESIGN . . . . .                   | 24 |
| A.   | DATABASE DESIGN . . . . .                  | 24 |
| B.   | PROGRAM DESIGN . . . . .                   | 24 |
| V.   | CONCLUSIONS . . . . .                      | 34 |
| A.   | INITIAL REACTIONS . . . . .                | 34 |
| B.   | SUMMARY . . . . .                          | 34 |
|      | APPENDIX A: SOURCE LISTINGS . . . . .      | 36 |

|   |     |
|---|-----|
| APPENDIX B: DATA DICTIONARY . . . . .       | 105 |
| A. TABLES IN THE DATABASE . . . . .         | 105 |
| B. COLUMNS IN ALPHABETICAL ORDER . . . . .  | 108 |
| C. COLUMN DESCRIPTIONS . . . . .            | 111 |
| D. ERROR MESSAGES . . . . .                 | 115 |
| <br>APPENDIX C: USER'S MANUAL . . . . .     | 118 |
| A. GETTING STARTED . . . . .                | 118 |
| 1. System Requirements. . . . .             | 118 |
| 2. Overview. . . . .                        | 118 |
| 3. Special Considerations. . . . .          | 119 |
| 4. Booting the System. . . . .              | 120 |
| 5. Exiting From R:base. . . . .             | 121 |
| B. SYSTEMS MANAGEMENT . . . . .             | 121 |
| 1. General. . . . .                         | 121 |
| 2. Adding Systems Records. . . . .          | 121 |
| 3. Editing Systems Data. . . . .            | 124 |
| 4. Deleting Systems Records. . . . .        | 126 |
| 5. The Systems Query Process. . . . .       | 127 |
| 6. Systems Management Reports. . . . .      | 130 |
| C. PROJECT MANAGEMENT . . . . .             | 131 |
| 1. General. . . . .                         | 131 |
| 2. Adding Project Records. . . . .          | 131 |
| 3. Editing Project Data. . . . .            | 135 |
| 4. Deleting Project Records. . . . .        | 137 |
| 5. The Project Query Process. . . . .       | 138 |
| 6. Project Management Reports. . . . .      | 140 |
| D. DATABASE MANAGEMENT . . . . .            | 140 |
| 1. General . . . . .                        | 140 |
| 2. Unloading Data to Send to HQMC . . . . . | 142 |
| 3. Merging Data from Remote Sites at HQMC . | 143 |
| 4. Packing the Database . . . . .           | 143 |
| 5. Archiving Records . . . . .              | 144 |
| 6. Removing Archived Records to History . . | 145 |

|                                       |     |
|---------------------------------------|-----|
| E. SUGGESTIONS AND COMMENTS . . . . . | 145 |
| LIST OF REFERENCES . . . . .          | 147 |
| BIBLIOGRAPHY . . . . .                | 148 |
| INITIAL DISTRIBUTION LIST . . . . .   | 151 |

## LIST OF TABLES

|      |  |     |
|------|--|-----|
| I    | TRADITIONAL SYSTEMS DEVELOPMENT TASKS . . . . .    | 21  |
| II   | BACHMAN DIAGRAM . . . . .                          | 25  |
| III  | STRUCTURE CHART MAIN LEVEL . . . . .               | 27  |
| IV   | STRUCTURE CHART SYSTEMS MANAGEMENT . . . . .       | 28  |
| V    | STRUCTURE CHART SYSTEMS QUERIES . . . . .          | 29  |
| VI   | STRUCTURE CHART SYSTEMS MANAGEMENT REPORTS . . . . | 30  |
| VII  | STRUCTURE CHART PROJECT MANAGEMENT . . . . .       | 31  |
| VIII | STRUCTURE CHART PROJECT QUERIES . . . . .          | 32  |
| IX   | STRUCTURE CHART PROJECT MANAGEMENT REPORTS . . . . | 33  |
| X    | SYSTEMS MANAGEMENT REPORTS . . . . .               | 132 |
| XI   | PROJECT MANAGEMENT REPORTS . . . . .               | 141 |

## LIST OF FIGURES

|       |   |     |
|-------|---|-----|
| C. 1  | MCASMIS Main Menu . . . . .                     | 121 |
| C. 2  | Systems Management Main Menu . . . . .          | 122 |
| C. 3  | Systems Management Add Menu . . . . .           | 122 |
| C. 4  | Systems Management Edit Menu . . . . .          | 124 |
| C. 5  | Systems Management Delete Menu . . . . .        | 126 |
| C. 6  | Systems Management Query Menu . . . . .         | 128 |
| C. 7  | Select Output Menu . . . . .                    | 128 |
| C. 8  | Systems Management Management Reports . . . . . | 131 |
| C. 9  | Project Management Main Menu . . . . .          | 133 |
| C. 10 | Project Management Add Menu . . . . .           | 133 |
| C. 11 | Project Management Edit Menu . . . . .          | 135 |
| C. 12 | Project Management Delete Menu . . . . .        | 137 |
| C. 13 | Project Management Query Menu . . . . .         | 138 |
| C. 14 | Project Management Management Reports . . . . . | 140 |
| C. 15 | Database Management Main Menu . . . . .         | 141 |

## I. INTRODUCTION

### A. PURPOSE

The life of a system starts with the identification of a deficiency during preliminary mission analysis. Marine Corps Order P5231.1 promulgates instructions regarding the management of all automated systems from the project initiation phase, through such time as that system is replaced or terminated. Systems analysts produce various development item documents, including a mission element need statement (MENS), and three system decision papers, prior to system approval and implementation. Additional configuration item documentation, such as system and program specifications, users manuals, etc., are produced and maintained after system implementation.

The assignment of document preparation tasks, and the subsequent retrieval of information regarding these documents, is a complex and tedious process. There is no central repository for this information. The manual process of retrieving information from multiple sources is time consuming and error prone.

The Marine Corps Automated Systems Management Information System (MCASMIS) is an integrated systems and project management tool. It was designed to facilitate the administration of all automated information systems centrally managed at the Headquarters, U. S. Marine Corps level.

The Manpower Systems Integration Branch, Code MPI-40 at Headquarters, Marine Corps, will be the primary user of this system for systems management. The Manpower Information Systems Liaison Unit (Code MPI-70), located in Kansas City, Missouri, will be co-users of the project management subsystem together with Code MPI-40. A system manager at

Headquarters, Marine Corps will perform all data administration aspects of this system.

MCASMIS will provide an automated storage and retrieval system for all systems documentation, while tracking the progress of systems development and providing information on milestone dates by means of exception reports. This will replace the current manual mode that relies primarily on personal notes and individual memories.

Various projects are opened for particular systems both during the development and maintenance portions of a systems life cycle. MCASMIS will record the contents, deliverables and implementation dates of these projects for later retrieval and report consolidation.

#### B. SCOPE

The MCASMIS programs were obviously tailored to this specific Marine Corps environment. Nevertheless, they may be easily modified and generalized to be useful for any staff involved in the systems management arena.

Because of the unique configuration problems associated with the distributed processing of project related information at both Kansas City and Washington, D.C., all the security features provided by the database management system could not be fully used. These problems will be discussed in more depth in Chapter 3.

The administration of all systems and projects requires vast financial resources. Unfortunately, time was not sufficient to include this financial aspect within the scope of this paper. It is a suggested area for future versions of this prototype system.

#### C. OUTLINE

Chapter two provides an overview of the operational environment. It discusses peculiar aspects of the project management tasks and delineates how these tasks are

currently performed. Finally, it outlines how MCASMIS will alleviate many problems with the current methodology.

Chapter three describes the database management software that was used to implement MCASMIS. It provides a critique of this software in terms of capabilities and limitations. It also discusses the design methodology that was used and briefly summarizes relative strengths and weaknesses of the prototyping technique.

Chapter four contains a Bachman diagram showing the relationships of the tables within Mcasmis, and a number of structure charts depicting the hierarchy of the programs.

Chapter five recaps the initial reaction of the users to the first version of the prototype system. It then summarizes the conclusions for the project.

## II. NATURE OF THE PROBLEM

### A. OPERATIONAL ENVIRONMENT

The Manpower Systems Integration Branch, Code MPI-40, is located at Headquarters, U.S. Marine Corps in Washington D.C. This branch centrally manages over one hundred automated information systems. All proposed manpower related systems are staffed through this office, and the life cycle documentation for these systems are administered and filed here. While some of these development item documents are produced within the Marine Corps, many others are contracted out to private industry. Processing sites for these systems are widely distributed. Most of the Class I Marine Corps systems are sponsored at one of three large data processing installations located in Kansas City, MO, Albany, GA, and Quantico, VA. Additional processing sites for these systems are located throughout the world.

The analysts from MPI-40 will be the only authorized users to modify the contents of the systems data in the MCASMIS database. Nevertheless, all systems data will be available for retrieval by other users as well. MPI-40 personnel also use the system to record projects that were opened by them for each of the systems. Many projects are opened for work by civilian contractors outside the Department of Defense. Thus, analysts must maintain close coordination with multiple development activities in addition to the activities operated and managed within the Marine Corps.

The Manpower Information Systems Liaison Unit (Code MPI-70) is located in Kansas City, MO. The Central Design and Programming Activity (CDPA) is co-located with the Marine Corps Finance Center (MCFC) in Kansas City and has primary responsibility for all automated systems that

administer the pay and personnel records of active duty, reserve and retired Marines. MPI-70 is the primary manpower functional representative for the following systems which are processed at this CDPA:

- JUMPS/MMS: an integrated manpower management and financial system for active duty Marines;
- REMMPS: an integrated manpower management and pay system for Marines in the Reserve establishment;
- ARMS: an automated recruit accession and management system; and
- CPMS: an automated career planning system for all Marines throughout the world.

MPI-70 personnel will use MCASMIS to record and produce reports on all projects opened for these systems. Analysts will also analyze production problems that surface in any of these projects.

## B. PROJECT MANAGEMENT

Analysts open projects for systems both during the development phase and after the system has been implemented. Each project has a project identification number issued by MPI-40. This identifier serves as the key for all information about that project and must be a unique value.

### 1. Projects for Systems Under Development

Users at MPI-40 open most of the projects for systems under development. Private industry development activities receive contracts to complete many of these projects. Each project has one or more deliverables associated with it. Information recorded on each project includes a scheduled due date for that deliverable. Predefined queries are available to flag those projects that are overdue, or will be coming due soon. MCASMIS aids management control and planning from a centralized perspective.

### 2. Projects During the Maintenance Phase

Most of the projects are opened for systems that have matured beyond the development phase. After systems have been implemented, projects that modify or enhance

existing logic are initiated for various reasons. A category is stipulated for each project at the time that project is opened. This category denotes both the urgency of the requirement, and the implementation method. Categories of projects opened are:

a. Production Modifications

Production Modifications are denoted by a P in the category field. These types of changes are normally too urgent to wait for a test cycle implementation. For example, erroneous logic that prevents a Marine's correct pay computation, or errors uncovered in the fallout of a test cycle implementation. This is the most dangerous type of modification since the chances of compounding the error are far greater than under circumstances where the modification is more thoroughly tested.

b. Test Cycle Modifications

Test Cycle Modification are denoted by a T in the category field. The CDPA implements two test cycles per year for the major systems under its sponsorship. All large changes that have no over-riding urgency are consolidated into a separate version of application programs. Quality assurance analysts then test this version before linking it to production. Most of the changes implemented at the CDPA fall into this category. This is the safest type of modification because of all quality assurance and change control standards are enforced before linking. Unfortunately, this type of change requires a long lead time and it is not always feasible.

c. Stand Alone Modifications

Stand Alone Modifications are denoted by an S in the category field. Some projects are too complex to handle as a production modification. They require additional safeguards and testing before they can be linked to production. Nevertheless, they may not be able to wait for the next test

cycle implementation for various reasons. For example, a legislatively imposed entitlements change, a GAO audit requirement, or other such large projects that have a predetermined deadline may fall into this category. These types of modifications are seldom used because they require analysts and programmers to maintain another version of all programs in addition to the production version and the test cycle version. It causes real nightmares for the applications programming personnel and the standards personnel who must enforce change control.

### 3. Project Priorities

Project information is input by MPI-70 users as well as users at MPI-40. Users assign each project a priority number. This enables the CDPA or other development activity to allocate the proper resources depending on the functional urgency. MCASMIS provides a multitude of management reports that enhance the control of these projects until they are completed.

## C. STATUS QUO

With the minor exception of a few standard reports that are produced by word processor, virtually all management tasks are performed manually with no automated support. Project folders are started for each project and system task. These folders are frequently misplaced. Much of the information about projects and systems exist only in the memories of the major participants and are not recorded. For example, participants negotiate agreements at project meetings, and fail to record these agreements in the project folder. Later recollections of those agreements frequently differ between the attending parties. More importantly, possible repercussions of project implementation fail to sift down to the operational level when these side effects are discussed at upper level meetings. Modifications to the original request are seldom documented formally. The

current control system places far too much emphasis on the memories of the respective analysts.

Typewritten reports concerning the status of various projects for the systems that are processed are produced periodically. These reports are then mailed to Headquarters from Kansas City. Ad hoc reports are time intensive and expensive to process. There is no central repository for all pertinent data. Subsequently, analysts spend many hours researching voluminous paper files to assimilate the answers to these requests.

The turnover of key personnel every two or three years further exacerbates the problem. This turnover completely strips the branch of invaluable corporate memory that is not easily recouped. Responsible parties miss milestones not out of neglect, but out of ignorance about the impending milestone date.

#### D. PROPOSED SYSTEM

MCASMIS seeks to automate much of the record keeping for all systems, development activities and projects. When a system development task is undertaken, all milestone dates are planned and input. MCASMIS then produces standard reports on the status of those documents. For example, which documents have not been tasked to anyone yet? Which milestones are coming due within the next X number of days? Which development activities have produced this type of documentation before? By entering just a few key strokes, the analyst can access the database in a manner such that a fast and accurate response to almost any ad hoc request is possible.

Management reports produced on request will greatly enhance the administration of all systems under the cognizance of the branch. Weak or potential problem areas are rapidly and automatically identified to permit the reallocation of personnel resources to effect timely resolution.

MPI-40 should designate a systems manager to oversee the data integrity of the MCASMIS database. The tasks of archiving records, reloading and packing the database, and unloading records to permanent history files on magnetic diskettes will fall under the purview of this system manager.

### III. METHODOLOGY

#### A. GENERAL

System and project management are inter-related functions. For example, projects are opened only for predefined systems. Deliverables are related to projects. Financial obligations are incurred for projects and systems. Database technology allows all of the data about projects and systems to be processed as an integrated whole. It provides program/data independence which permits flexible and powerful ad hoc retrieval that is not easily supported by file processing systems. Any future changes to the structure of the data does not require that the programs which access this data also be modified. It eliminates data duplication. For example, data about projects need not be repeated separately in project and system application programs.

Headquarters selected R:base 5000, a micro-oriented, relational database management system, systems development. The MCASMIS system was developed on an IBM PC-XT, and should run on all completely compatible equipment. It was tested on a Zenith Z-150 and ran satisfactorily on that machine. MCASMIS requires a hard disk.

#### 1. Capabilities of R:base

##### a. Programming Language

R:base comes with a programming language that is well documented and easy to learn. All structured constructs are supported and the programs are easily modularized to produce efficient, robust and maintainable code. Screen design is facilitated by specific commands that generate bordered menus and on-line help screens. R:base includes a menu-driven, automatic code generator which permits flexible data retrieval for even a novice user.

b. Compiler

A built in compiler converts ASCII application program files to binary representation. This not only provides additional processing speed, but helps preserve the integrity of the programs. The user can neither view, nor modify, this source code with text editors.

c. Data Dictionary

A passive data dictionary provides on-line documentation of the table and column names within the database. R:base automatically maintains key values, and verifies the format according to predefined item attributes. For example, types defined as date, dollar, time and integer are automatically checked for proper format at time of entry.

d. Automatic Rule Checking

R:base permits the identification of rules for data entry. Subsequent attempts to load or modify a record that violates any of these rules are automatically rejected by the system. This feature is invaluable in preserving data integrity from a global perspective. It eliminates the requirement to code for this editing within the separate programs. For example, a rule might be defined which prohibits opening a project for a non-existent system. Any attempt to load a project that violates this rule would be rejected.

2. Limitations of R:base

a. Security

R:base permits the assignment of passwords for writing or reading data from individual tables. Unfortunately, these passwords can only be defined if there is an owner password defined for the entire database. However, when an owner password is assigned, only that user is allowed to unload data from the database. This is unacceptable in the current situation. Data must be unloaded at both Kansas City and Washington to keep the database

current. Therefore, MCASMIS depends on manual procedures to preserve the data security. R:base does not provide the flexibility to assist in this unique configuration environment.

b. Name Scoping

All variables defined in R:base are global. The principle of information hiding is violated by not limiting the visibility of the variables according to the structure of the program in terms of subordinate modules.

Furthermore, R:base permits, but does not require, the explicit declaration of variables before use. A stronger typing would prevent cumbersome run time errors, and would enforce traditionally accepted standards of structured programming.

B. DESIGN METHODOLOGY

1. Defining Requirements

Typically the hardest task associated with systems development has been to adequately define the requirements. Table I depicts the series of tasks performed in the traditional systems development process.

TABLE I  
TRADITIONAL SYSTEMS DEVELOPMENT TASKS

Feasibility Study  
Requirements Definition  
Preliminary Design  
Detailed Design  
Coding  
Integration  
Implementation  
Operations and Maintenance

These steps normally result in voluminous documentation that is rarely understood by any one person in its

entirety. It is also a lengthy process which runs the risk of users changing their minds before completion. Even when the users are involved in approving the results of each task, the outcome cannot be assured. Often, the users feel coerced into signing off on something that is not completely accurate, or even understood. Yet, they know that they will not get to see anything concrete until further phases have been completed. It is not uncommon for the users to react to the resultant system with surprise and disdain. One overheard response was, "I do not know what I want, but I do know that that's not it!" Even with the exceptionally computer literate users of MCASMIS, the traditional approach to systems development was infeasible for this project. A two day visit was made to MPI-40 to interview the analysts, obtain insight into their tasks and responsibilities, and collect sample output reports that would be desirable. All personnel were anxious to help and answered all questions. Nevertheless, it was difficult to specifically itemize all requirements of the proposed system. Tasks were sufficiently unstructured to preclude exact descriptions of duties and unambiguous user specifications.

It was therefore decided that the best course of action would be to build a functional model of a proposed system. This prototype would seek to alleviate many of the present problem areas in the systems management arena.

## 2. Prototyping

In contrast to the rigid, disciplined approach to systems development using the structured analysis methodology, prototyping is an informal procedure that results in faster convergence between need and specification. Specifications are refined as users interact with the prototype system and the analyst who developed that system. Many experts are recognizing the advantages of such a methodology for decision support systems. As described in [Ref. 1],

A structured transaction-oriented system requires an intensive up-front design effort to get firm specifications which can then be programmed. A decision support system, however, involves more a process of user learning. An appropriate methodology here is often a crude design followed by a simple program. Use of the programs by the user leads to successively different and more comprehensive design as its performance is analyzed and then to a series of new programs. Interactively, one cycles through this sequence a number of times. Such a design process (pragmatically useful) flies in the face of generally held nostrums about good development practices.

The geographic separation from the intended users also contributed to the decision to use the prototyping approach. Without detailed written specifications, this methodology offered the highest probability of ultimately satisfying the system users.

The main advantages of using prototyping are, a better understanding of user requirements through faster feedback, and the potential for using the prototype as the final solution [Ref. 2]. It is this second advantage that also contributes to a serious pitfall of this methodology. If the prototype is accepted as the final solution, there is danger of incorporating logical errors that will render subsequent enhancement unattainable. A prototypic design is almost purely physical in nature. An unambiguous statement of specifications from a logical perspective would preclude such errors from being perpetuated in the maintenance phase. Fortunately, R:base provides a flexible array of data manipulation functions that should accommodate any necessary schema modifications should a logical error surface. Of course, this capability is not unique to R:base. It is an inherent advantage of all relational data base management systems which provide program/data independence.

#### IV. MCASMIS DESIGN

##### A. DATABASE DESIGN

Table II depicts the logical relationships of the primary tables in the MCASMIS database. Also included in this database are copies of the system, project, TTC and deliver tables that are used for temporary storage of archived records. The number of arrows shown for each table on this Bachman Diagram denote either a one to many or a many to many relationship. For example, each project could have many deliverables, but every deliverable is associated with only one project. Conversely, each project could effect multiple TTCs, and each TTC could be related to many projects.

The relationships between the files are connected by key fields which must be defined as unique values. The MPI-ID serves as the key which establishes the relationship between project, TTC and deliverable records. The SYSNAME is the key value for system and doclist records, and DEVACT is the key for vendor records.

##### B. PROGRAM DESIGN

Tables III through IX show the hierarchical structure of the application programs that comprise MCASMIS. The system was designed to be completely user friendly. On-line help is available at any of the menu screens by pressing F10. MCASMIS also provides an emergency escape mechanism to return directly to the main menu from any of the subordinate modules.

Table III depicts the main level of the program as well as the modules that comprise the database management aspects of MCASMIS.

TABLE II  
BACHMAN DIAGRAM

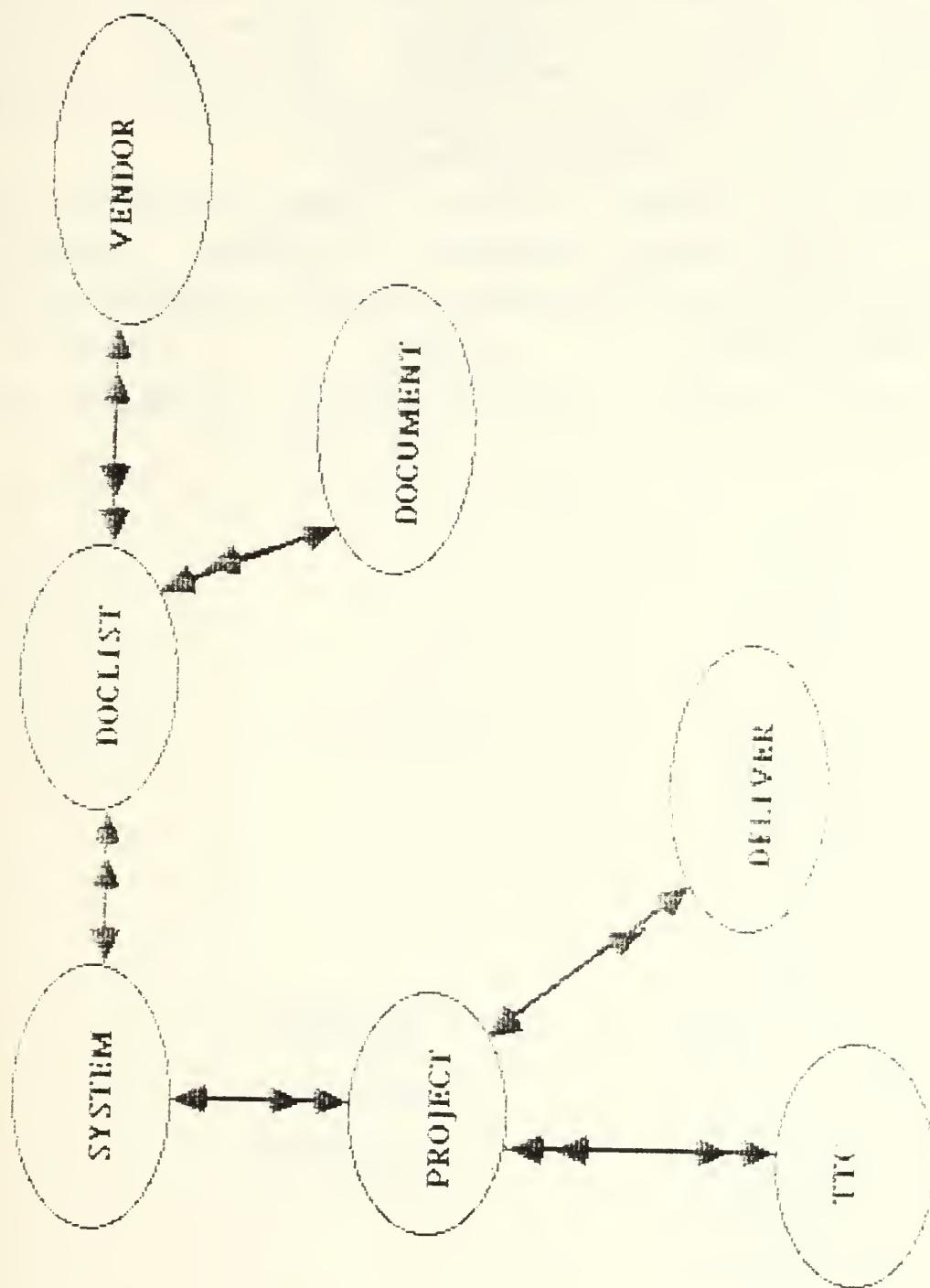


Table IV shows the standard add, change and delete functions for systems management. Tables V and VI display the hierarchy for the systems management query and management reports options respectively. Queries are normally one of a kind, ad hoc requests that may be used to answer many questions that arise in the course of systems administration. Management reports are usually more structured, recurring data retrievals that provide management control and assist in the planning of resource allocation.

Tables VII through IX depict the same respective functions for the project management subsystem. Appendix C contains detailed instructions on the use and purpose of all the above programs.

TABLE III  
STRUCTURE CHART  
MAIN LEVEL

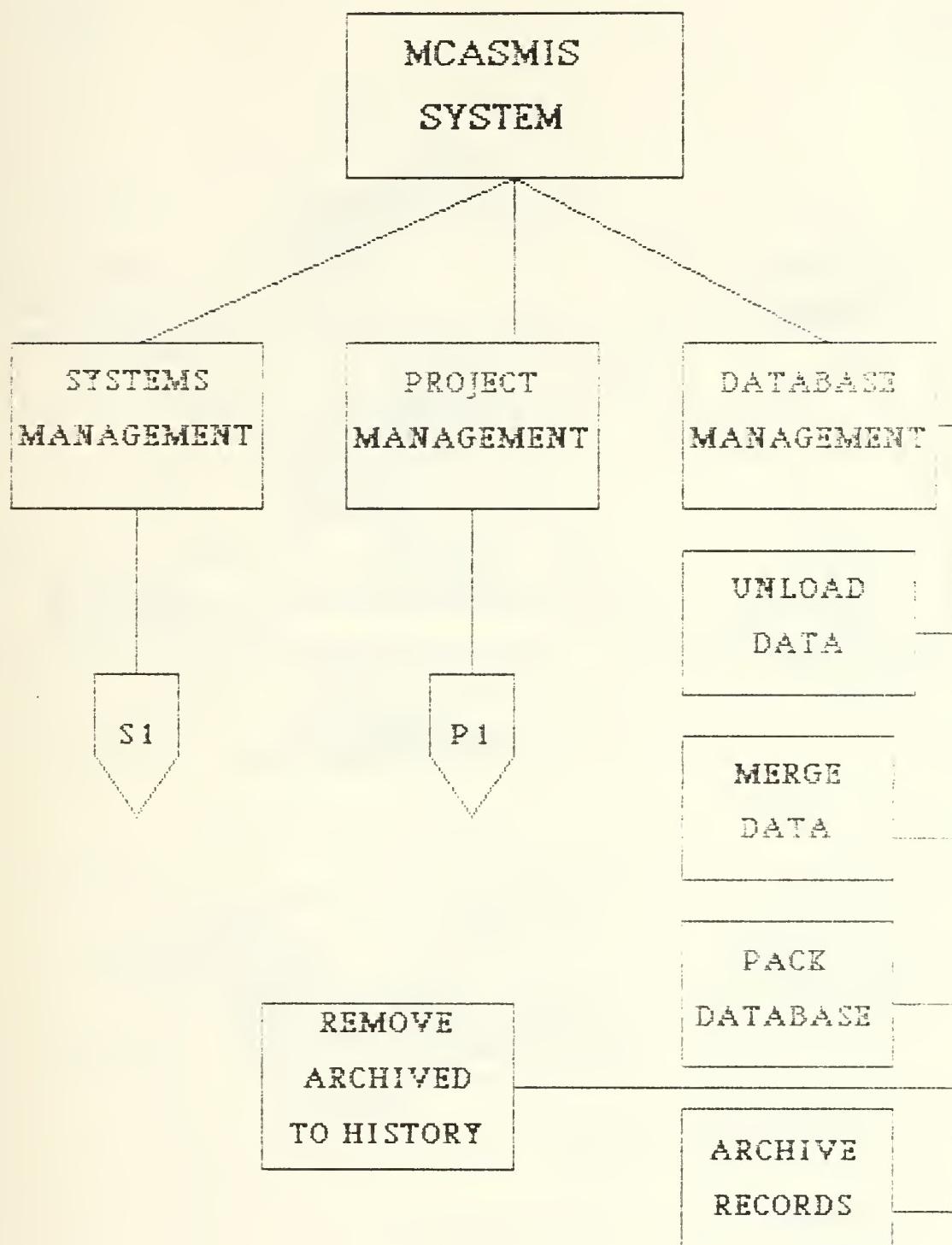


TABLE IV  
STRUCTURE CHART  
SYSTEMS MANAGEMENT

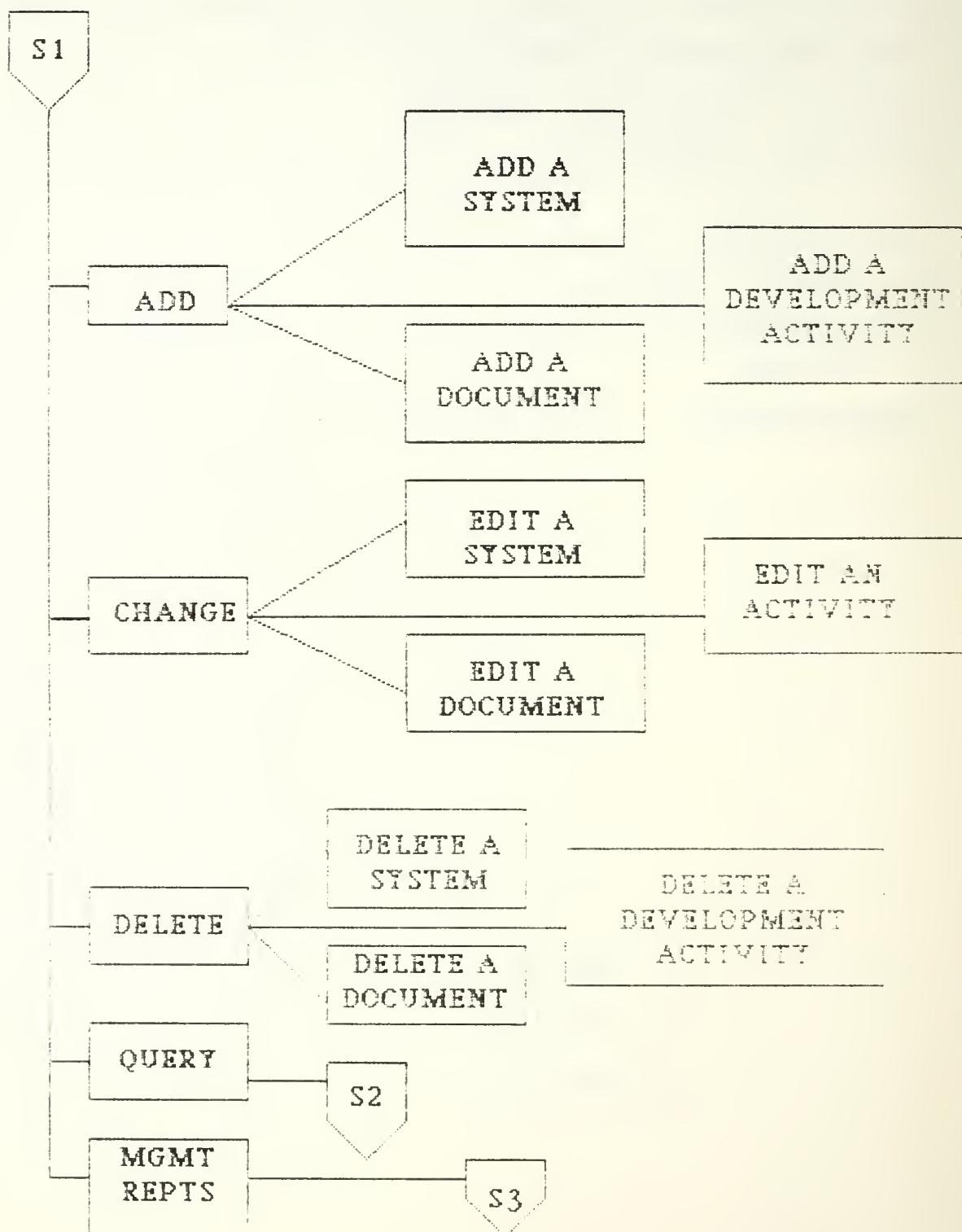


TABLE V  
STRUCTURE CHART  
SYSTEMS QUERIES

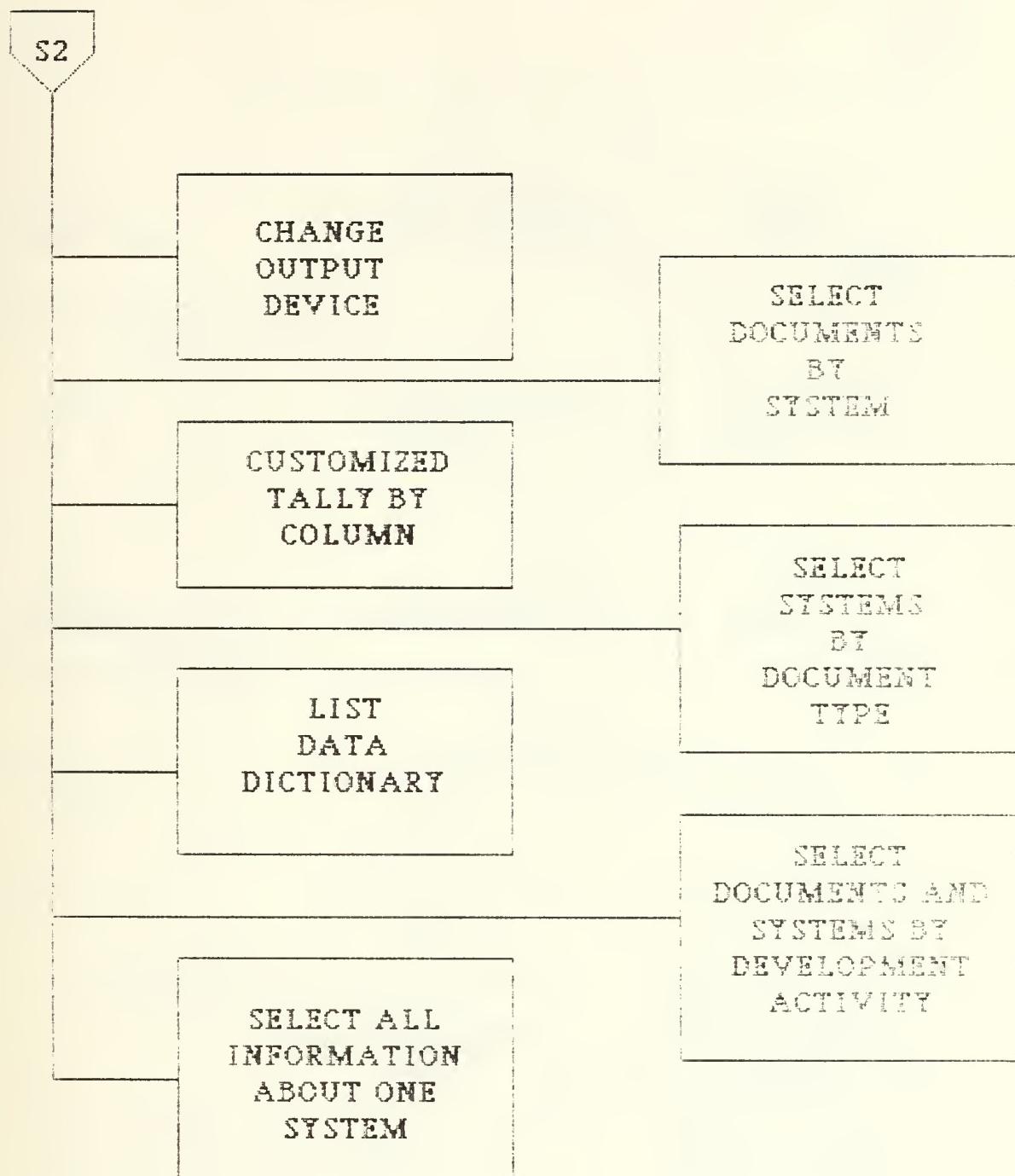


TABLE VI  
STRUCTURE CHART  
SYSTEMS MANAGEMENT REPORTS

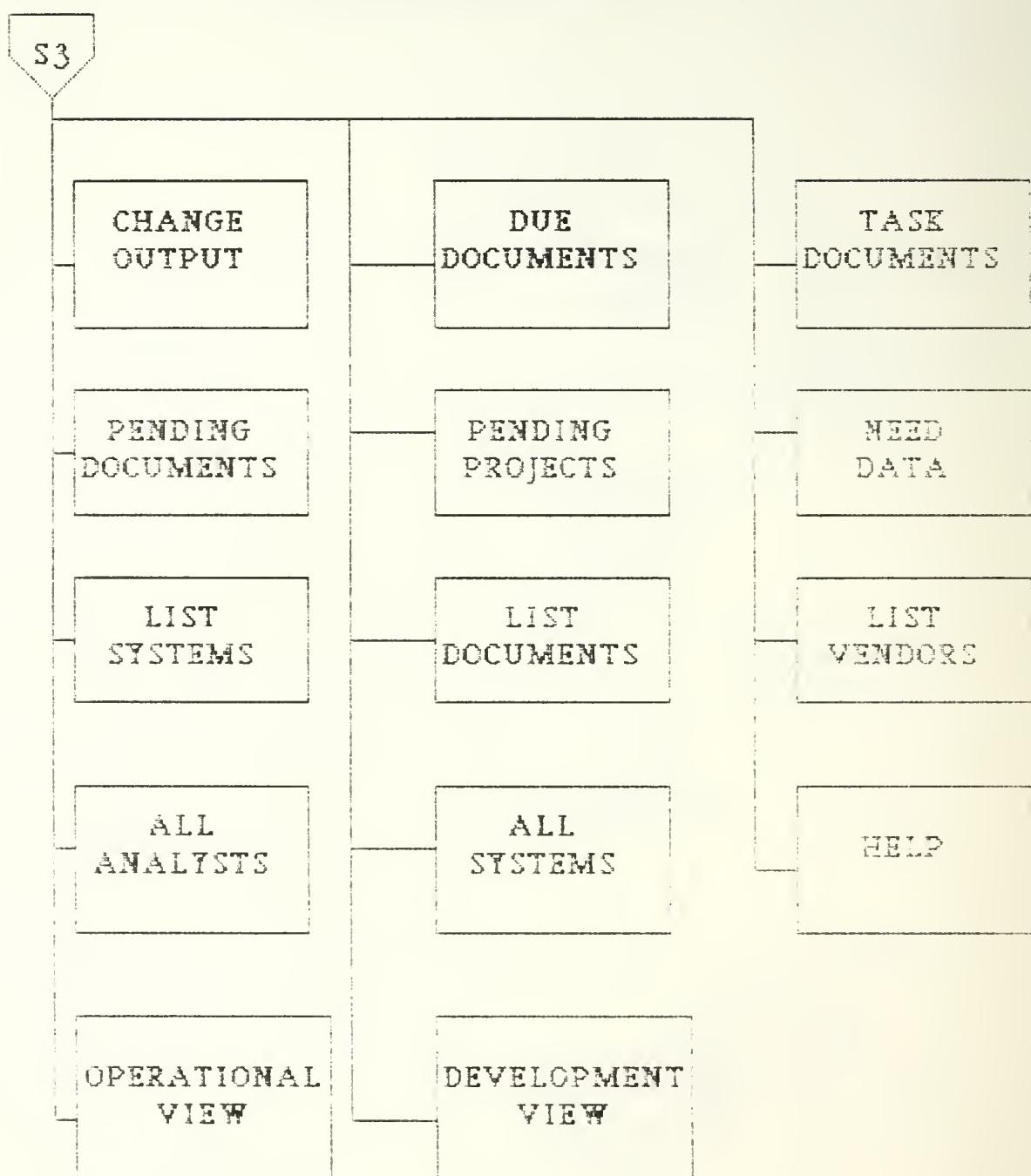


TABLE VII  
STRUCTURE CHART  
PROJECT MANAGEMENT

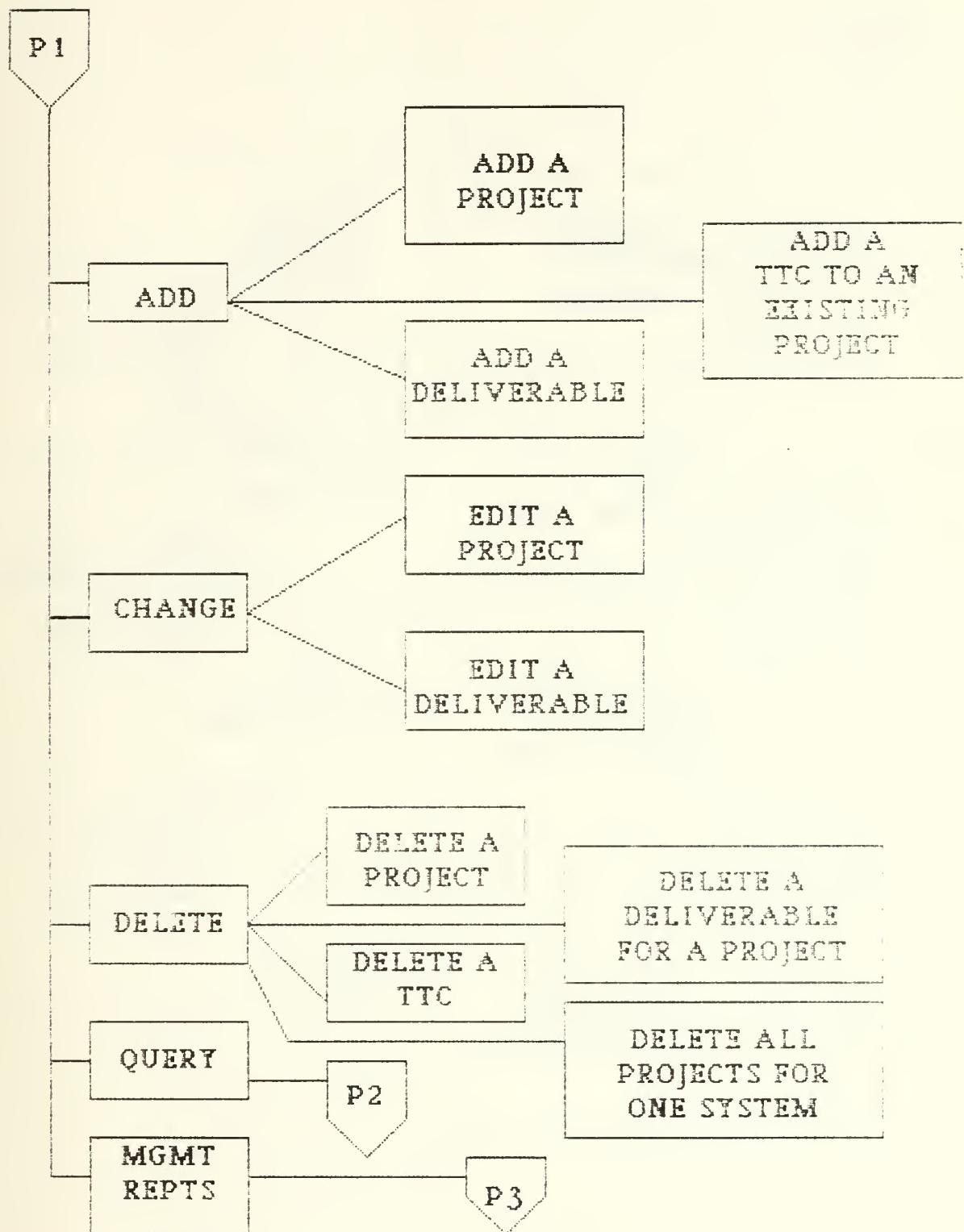


TABLE VIII  
STRUCTURE CHART  
PROJECT QUERIES

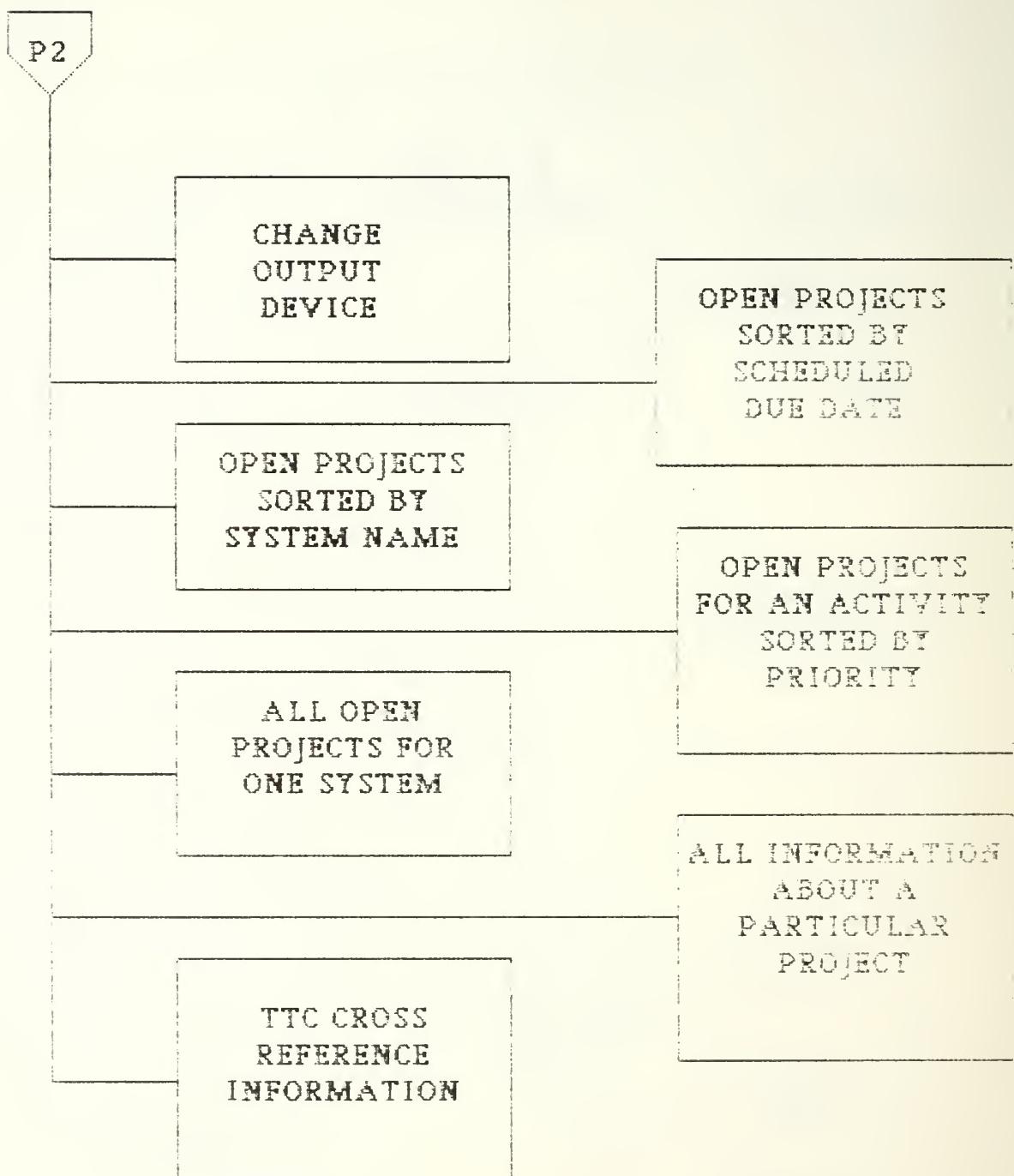
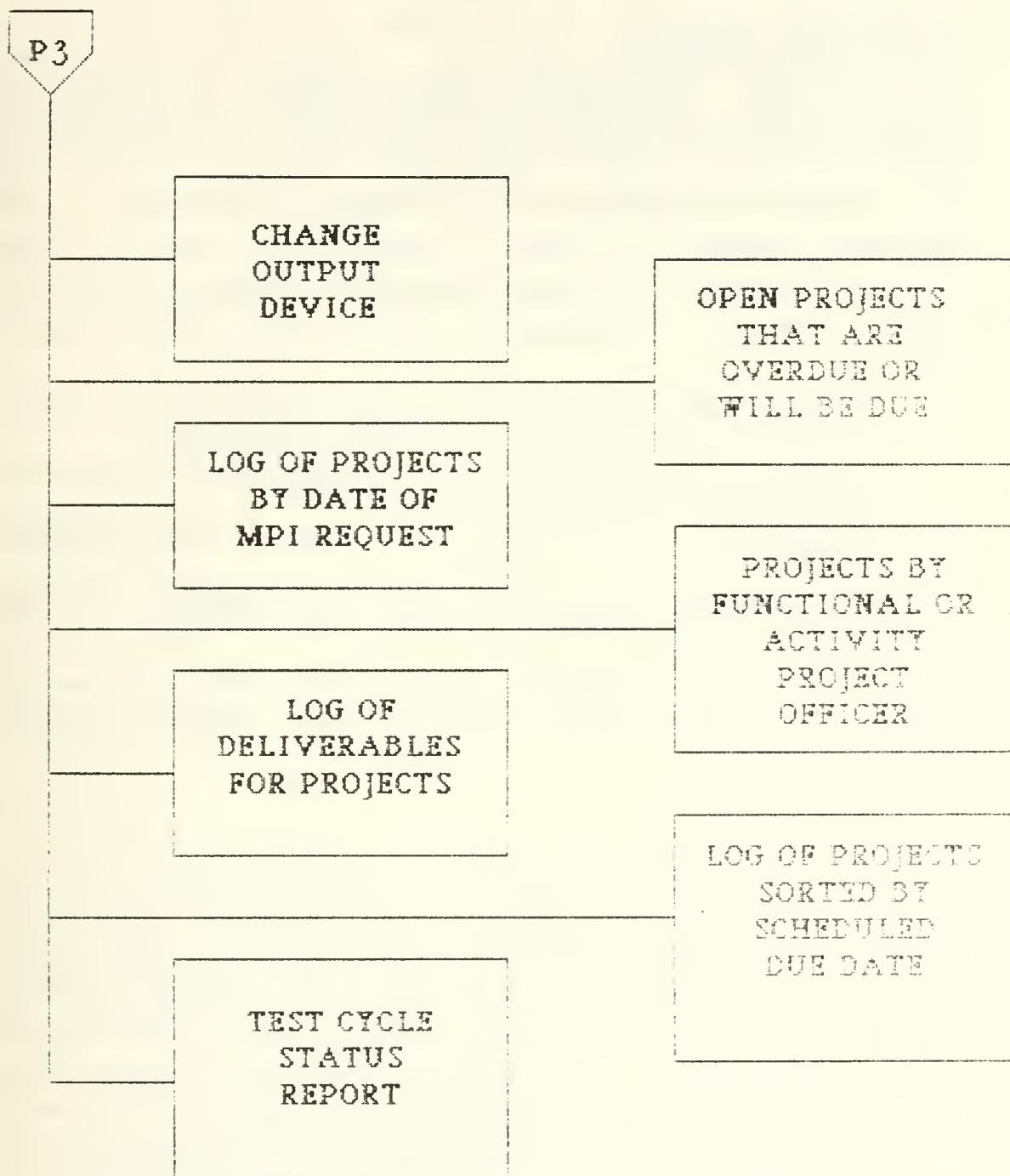


TABLE IX  
STRUCTURE CHART  
PROJECT MANAGEMENT REPORTS



## V. CONCLUSIONS

### A. INITIAL REACTIONS

It required approximately three months to develop the original version of the prototype system. A meeting to demonstrate this model was arranged in Kansas City. The user operated the system with virtually no assistance. The prototype proved to be user friendly and robust. This review uncovered a few human interface problems which were trivial to modify. Examples of some of the problems uncovered were:

- Providing advisory messages to the user that the application was working during lengthy retrievals;
- A less disruptive method for routing desired output to the printer as well as to the screen;
- A different sort criterion for one of the management reports; and
- Provide the option of listing valid arguments at some of the system prompts when the user is unsure of the input.

Some of the programs were also broken down into smaller modules. This noticeably increased the responsiveness of the system.

### B. SUMMARY

This thesis does not purport to recommend prototyping as an alternative to the rigorous, time-intensive structured systems development techniques. Nevertheless, prototyping did enable the production of a functional decision support system that could serve as the framework for a detailed set of user requirements. It could be used in its present form immediately. Feedback provided by the users will serve as an excellent and credible definition of requirements for system refinements. It is beyond the scope of this thesis to analyze the relative advantages and disadvantages of prototyping in general. Prototyping as a design methodology

has proved to be exceptionally effective in this particular development effort.

Future enhancements to MCASMIS should include the incorporation of financial resource data that are connected to each system and project, and more stringent data security regarding both passwords and procedures. The multi-user version of R:base may offer some assistance in this area.

APPENDIX A  
SOURCE LISTINGS

```
*( ====== PROGRAM: ADDTTC.CMD
      AUTHOR: R. C. CAVALLARO
      DATE WRITTEN: 29 DEC 1985
      PURPOSE: ADDS TTCS TO A NEW OR EXISTING PROJECT
TABLE USED: TTC
VARIABLE FORM USED: TTC
=====
SET VAR AGAIN = Y
SET NULL " "
WHILE AGAIN = Y THEN
  NEWPAGE
  WRITE "ENTER TO ADD NEW TTC FOR THIS PROJECT" AT 1 1
  DRAW TTC WITH ALL
  ENTER VAR VTTC RETURN ENTER ESC
  IF #RETURN = ESC THEN
    BREAK
  ENDIF
  LOAD TTC
    . VMID . VTTC
  END
  CLEAR VTTC
ENDWHILE
SET NULL -O-
RETURN
=====
*( ====== PROGRAM: DOOPICK.CMD
      AUTHOR : R. C. CAVALLARO
      DATE WRITTEN: 3 FEBRUARY 1986
      PURPOSE: THE MAIN MENU OF THE DATABASE MANAGEMENT SUBSYSTEM. ROUTES
                USER TO PROPER MENU BASED ON INDICATED INPUT TO MENU.
                ONLY THE DATABASE ADMINISTRATOR SHOULD EXECUTE OPTIONS
                2 THROUGH 5.
PROGRAMS CALLED: D01PICK.CMD, D02PICK.CMD, D03PICK.CMD, D04PICK.CMD,
                  D05PICK.CMD
SCREENS DISPLAYED: DOOHLIP, DMENU00
```

```

*(initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
    NEWPAGE
    WRITE "SELECT ANY OPTION WITH NUMBER OR CURSOR ARROW AND ENTER"
    WRITE "F10 FOR HELP AT 1 66"
    WRITE "CHOOSE CHOICE FROM DMENU00 PLEASE WAIT."
    WRITE "INPUT ACCEPTED . . PLEASE WAIT."
    IF CHOICE = -1 THEN          *( F10 was pressed for help )
        DISPLAY DOOHLP
        WRITE ANY KEY TO CONTINUE . . .
        PAUSE
        SET VAR CHOICE = 0
    ENDIF
    IF CHOICE = 1 THEN          *( add menu )
        RUN D01PICK.CMD
    ENDIF
    IF CHOICE = 2 THEN          *( edit menu )
        RUN D02PICK.CMD
    ENDIF
    IF CHOICE = 3 THEN          *( delete menu )
        RUN D03PICK.CMD
    ENDIF
    IF CHOICE = 4 THEN          *( query menu )
        RUN D04PICK.CMD
    ENDIF
    IF CHOICE = 5 THEN          *( management reports menu )
        RUN D05PICK.CMD
    ENDIF
    IF CHOICE = 6 THEN          *( return to MCASMIS main menu )
        BREAK
    ENDIF
    SET VAR CHOICE = 0
ENDWHILE
SET VAR CHOICE = 0
RETURN
*( reset loop control variable )

```

Option 1 will unload all information pertaining to projects which distributed

Sites have input 2 for subsequent transmittal to HQMC.  
 Option 2 enables HQMC to merge the files that were transmitted by distributed sites into the main database, reloads the database to recover space, and creates a diskette copy of the database to send to those sites.  
 Option 3 packs the database to recover space from records that have been deleted.  
 Option 4 archives project and document records in order to speed up the access to the active database.  
 Option 5 removes all archived records to a separate history diskette to conserve space on the hard disk.

```
* ( ======)
* PROGRAM: DO1PICK.CMD
* AUTHOR : R. C. CAVALLARO
* DATE WRITTEN: 3 FEBRUARY 1986
* PURPOSE: UNLOADS DATA FOR TTC, PROJECT AND DELIVERABLES FOR SUBSEQUENT
*           TRANSMITTAL TO HQMC.

PROGRAM CALLED: MAIN.CMD
=====)
=====)

NEWPAGE
SET VAR DRV = D
WHILE DRV NE A AND DRV NE B THEN
    FILLIN DRV USING "WHICH DRIVE WILL CONTAIN THE OUTPUT DATA: "
    IF DRV NE A AND DRV NE B THEN
        WRITE ""
        WRITE "YOU MUST ENTER EITHER A OR B AS VALID DRIVE DESIGNATORS."
        WRITE ""
    ENDIF
ENDWHILE
SET VAR DRV = :DRV + "OUT#DATA"
SET VAR DRV = .DRV + "OUT#DATA"
NEWPAGE
WRITE "WORKING . PLEASE STANDBY." AT 10 25
OUTPUT DRV
UNLOAD DATA FOR TTC
UNLOAD DATA FOR DELIVER
UNLOAD DATA FOR PROJECT
OUTPUT SCREEN
CLEAR DRV
QUIT TO MAIN.CMD
* ( return to MCASMSI main menu )

* ( ======
* PROGRAM: DO2PICK.CMD
* AUTHOR : R. C. CAVALLARO
* DATE WRITTEN: 3 FEBRUARY 1986
```

PURPOSE: USED BY DATABASE ADMINISTRATOR TO MERGE FILES TRANSMITTED BY REMOTE SITES INTO MAIN DATABASE. DELETES DUPLICATES, RECOVERS UNUSED SPACE AND CREATES A COPY OF THE DATABASE TO SEND BACK TO THE DISTRIBUTED SITES.

```
=====
NEWPAGE
SET ERR VAR ERV
SET VAR DRV = X
WHILE DRV NE A AND DRV NE B THEN
  FILLIN DRV USING WHICH DRIVE SHOULD CONTAIN THE DATA TO BE MERGED: "
  IF DRV NE A AND DRV NE B THEN
    WRITE "YOU MUST ENTER EITHER A OR B AS VALID DRIVE DESIGNATORS. "
    WRITE "
  ENDIF
ENDWHILE
SET VAR DRV = .DRV + \OUT#DATA"
NEWPAGE
WRITE WORKING . . PLEASE STANDBY. " AT 10 25
INPUT .DRV
DELETE DUPLICATES FROM PROJECT
DELETE DUPLICATES FROM TTC
DELETE DUPLICATES FROM DELIVER
NEWPAGE
SET VAR DRV = X
WHILE DRV NE A AND DRV NE B THEN
  WRITE "WHICH DRIVE SHOULD CONTAIN THE DATA TO RETURN TO THE REMOTE"
  FILLIN DRV USING SITES:
  IF DRV NE A AND DRV NE B THEN
    WRITE "YOU MUST ENTER EITHER A OR B AS VALID DRIVE DESIGNATORS. "
    WRITE "
  ENDIF
ENDWHILE
NEWPAGE
WRITE WORKING PLEASE STANDBY. "
SET VAR DRV = .DRV + \MCWIS
SET VAR DRV = :DRV + \MCWIS
RELOAD DRV
IF ERV NE O THEN
  WRITE "
  WRITE "THE RELOAD DID NOT WORK. TRY AGAIN OR CONTACT THE DBA. "
  WRITE "PRESS ANY KEY TO CONTINUE . "
  PAUSE
RETURN
ENDIF
```

```

SET VAR DRV = .DRV + ".*.RBS" *( close database for copying )
CLOSE MCMIS
COPY .DRV
OPEN MCMIS
CLEAR DRV
RETURN

*( =====
PROGRAM: DO3PICK.CMD
AUTHOR : R. CAVALLARO
DATE WRITTEN: 3 FEBRUARY 1986
PURPOSE: PACKS THE DATABASE AND RELOADS IT TO ENSURE THAT ALL SPACE THAT
WAS PREVIOUSLY USED BY DELETED RECORDS IS AGAIN AVAILABLE.
=====)

SET ERR VAR ERV *( set error variable )
NEWPAGE
SET VAR DRV = X
WHILE DRV NE A AND DRV NE B AND DRV NE C THEN
  FILLIN DRV USING WHICH DRIVE WILL BE USED FOR THE SCRATCH FILES:
  IF DRV NE A AND DRV NE B AND DRV NE C THEN
    WRITE ""
    WRITE "YOU MUST ENTER A, B OR C AS VALID DRIVE DESIGNATORS."
    WRITE
  ENDIF
ENDWHILE
NEWPAGE
WRITE "WORKING PLEASE STANDBY."
SET VAR DRV = .DRV +
SET VAR DRV = .DRV + ;SCRT#
RELOAD .DRV
IF ERV NE "O" THEN
  WRITE "
  WRITE "THE PACK HAS FAILED. TRY AGAIN OR CONTACT THE DBA."
  WRITE "PRESS ANY KEY TO CONTINUE . "
  PAUSE
RETURN
ENDIF
NEWPAGE
WRITE "RELOADING THE NEW DATABASE. PLEASE STANDBY." AT 12 25
SET VAR DRV = .DRV + ;RBS *( close database for copying )
CLOSE MCMIS
COPY .DRV MCMIS*.RBS *( reopen database )
OPEN MCMIS
ERASE .DRV
RETURN

```

```

*( =====
PROGRAM: DO4PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 3 FEBRUARY 1986
PURPOSE: ARCHIVES EITHER DOCUMENTS OR PROJECTS (WITH ASSOCIATED ITC AND
          DELIVERABLE RECORDS) TO SPEED UP ACCESS FOR ACTIVE RECORDS IN THE
DATABASE.

PROGRAMS CALLED: D14PICK.CMD, D24PICK.CMD, MAIN.CMD
SCREENS DISPLAYED: DO4HLP, DMENU04

=====
=====

*( initialize loop control variable
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
NEWPAGE
WRITE "SELECT ANY OPTION WITH NUMBER OR CURSOR ARROW AND ENTER"
WRITE "F1 FOR HELP" AT 1 66
WRITE "H" AT 1 66
CHOOSE "CHOICE FROM DMENU04"
WRITE "INPUT ACCEPTED . . PLEASE WAIT. "

IF CHOICE = -1 THEN
DISPLAY "DO4HLP"
WRITE "PRESS ANY KEY TO CONTINUE . . "
PAUSE
ENDIF

IF CHOICE = 1 THEN
*( archive development documents )
RUN D14PICK.CMD
ENDIF

IF CHOICE = 2 THEN
*( archive projects )
RUN D24PICK.CMD
ENDIF

IF CHOICE = 3 THEN
*( return to db main menu )
BREAK
ENDIF

IF CHOICE = 4 THEN
*( return to MCASMSI main menu )
QUIT TO MAIN.CMD
ENDIF

SET VAR CHOICE = 0
ENDWHILE
CLEAR ALL VAR

```

```
SET VAR CHOICE = 0  
RETURN
```

Option 1 archives development item documents for a specific system so that the access to the active database records will be speeded up. Once these records are archived, they will no longer appear in any normal retrievals from MCAS MIS, but are still available through the DBA. Option 2 will perform the same function as above, but for projects with the associated TTC and deliverable data on those projects.

```
*(*=====PROGRAM: DOSPICK.CMD  
AUTHOR: R.C. CAVALLARO  
DATE WRITTEN: 3 FEBRUARY 1986  
PURPOSE: REMOVES ALL ARCHIVED RECORDS FROM THE DATABASE HISTORY TABLES  
AND TRANSFERS THEM TO A DISKETTE FOR POSTERITY. THIS CLEARS  
UP SPACE ON THE HARD DISK THE NEXT TIME THE DATABASE IS PACKED.  
=====)  
  
SET VAR DRV = X  
WHILE DRV NE A AND DRV NE B THEN  
  FILLIN DRV USING "WHICH DRIVE WILL THE DATA BE OUTPUT TO:  
  IF DRV NE A AND DRV NE B THEN  
    WRITE " "  
    WRITE "ENTER EITHER A OR B AS VALID DRIVE DESIGNATORS."  
    WRITE " "  
    WRITE "PRESS ANY KEY TO CONTINUE . ."  
    PAUSE  
  ENDIF  
ENDWHILE  
SET VAR DRV = X  
FILLIN ENAME "+WHAT IS THE NAME OF THE FILE ( e. g. CALYR85): "  
SET VAR DRV = .DRV + .ENAME  
NEWPAGE  
WRITE "WORKING . . PLEASE STANDBY."  
OUTPUT DRV  
UNLOAD DATA FOR DOCHIST  
UNLOAD DATA FOR PROJHIST  
UNLOAD DATA FOR TTCHIST  
UNLOAD DATA FOR DLVRHIST  
OUTPUT SCREEN  
DELETE ROWS FROM DOCHIST WHERE DOCABR EXISTS  
DELETE ROWS FROM PROJHIST WHERE MPI-ID EXISTS  
DELETE ROWS FROM TTCHIST WHERE MPI-ID EXISTS  
DELETE ROWS FROM DLVRHIST WHERE MPI-ID EXISTS  
CLEAR ALL VAR  
SET VAR CHOICE = 0
```

RETURN

```
* ( ====== PROGRAM: D14PICK.CMD
      AUTHOR: R. C. CAVALLARO
      DATE WRITTEN: 3 FEBRUARY 1986
      PURPOSE: ARCHIVES DEVELOPMENT ITEM DOCUMENT RECORDS FOR A SYSTEM AFTER
              THAT SYSTEM HAS BEEN IMPLEMENTED.
===== )
```

```
      WRITE "H FILLIN V$YS USING "WHAT IS THE SYSTEM ABBREVIATION:
      SET POINTER #1 PT1 FOR DOCLIST WHERE SYSNAME = .V$YS
      IF PT1 NE 0 THEN
          WRITE "THERE ARE NO DOCUMENTS ON FILE FOR THAT SYSTEM."
          WRITE "W
          WRITE "PRESS ANY KEY TO CONTINUE . .
          PAUSE
          RETURN
```

```
      ENDIF
      SET POINTER #1 PT1 FOR DOCUMENT WHERE LIFECAT = DI
      WHILE PT1 = 0 THEN
          SET VAR ITEM = DOCABR IN #1
          * use additional search variables for multiple document types )
          SET VAR ITEM1 = "ITEM +
          SET VAR ITEM2 = "/ + .ITEM
          SET VAR MSG TEXT
          SET VAR MSG = "ARCHIVING RECORDS OF TYPE" & .ITEM
          SHOW VAR MSG
          APPEND DOCLIST TO DOCHIST WHERE DOCABR = .ITEM AND SYSNAME = .V$YS +
          OR DOCABR CONTAINS ITEM1 AND SYSNAME = V$YS OR +
          DOCABR CONTAINS ITEM2 AND SYSNAME = V$YS
          DELETE ROWS FROM DOCLIST WHERE DOCABR = .ITEM AND SYSNAME = .V$YS +
          OR DOCABR CONTAINS ITEM1 AND SYSNAME = V$YS OR +
          DOCABR CONTAINS ITEM2 AND SYSNAME = .V$YS
          NEXT #1 PT1
      ENDWHILE
      CLEAR ALL VAR
      SET VAR CHOICE = 0
      RETURN
```

```
* ( ====== PROGRAM: D24PICK.CMD
      AUTHOR: R. C. CAVALLARO
      DATE WRITTEN: 3 FEBRUARY 1986
      PURPOSE: ARCHIVES PROJECT RECORDS WITH ASSOCIATED TTC AND DELIVERABLE
              RECORDS THAT WERE COMPLETED AFTER A SPECIFIED DATE.
```

```

SET ERR VAR ERV
SET VAR TGT DATE
LABEL LOOP
NEWPAGE
WRITE "PLEASE INPUT THE DATE THAT YOU WANT TO KEEP RECORDS IN THE ACTIVE"
WRITE "DATABASE. ALL PROJECTS COMPLETED BEFORE THIS DATE WILL BE ARCHIVED"
FILLIN TGT USING "AND CAN ONLY BE RETRIEVED BY THE DBA:
IF ERV NE 0 THEN
WRITE " "
WRITE "YOU MUST INPUT A VALID DATE IN DD MMM YYYY FORMAT"
WRITE " "
WRITE "PRESS ANY KEY TO CONTINUE . . "
PAUSE
GOTO LOOP
ENDIF
NEWPAGE
SET POINTER #1 PT1 FOR PROJECT WHERE COMPLETE LT . TGT
IF PT1 NE 0 THEN
WRITE "NO PROJECTS WERE COMPLETED BEFORE THIS DATE"
WRITE "PRESS ANY KEY TO CONTINUE . . "
PAUSE
ELSE
WHILE PT1 = 0 THEN
SET VAR MID = MPI-ID IN #1
SET VAR MSG TEXT
SET VAR MSG = "ARCHIVING RECORDS FOR PROJECT" & .MID
SHOW VAR MSG
APPEND TTC TO TCHLIST WHERE MPI-ID = MID
DELETE ROWS FROM TTC WHERE MPI-ID = MID
APPEND DELIVER TO DLVRHIST WHERE MPI-ID = MID
DELETE ROWS FROM DELIVER WHERE MPI-ID = MID
APPEND PROJECT TO PROJHIST WHERE MPI-ID = MID
DELETE ROWS FROM PROJECT WHERE MPI-ID = MID
NEXT #1 PT1
ENDWHILE
ENDIF
CLEAR ALL VAR
SET VAR CHOICE = 0
RETURN

```

```

*( ====
PROGRAM: DELIVADD.CMD
AUTHOR: R. C. CAVALLARO
DATE WRITTEN: 29 DEC 1985
PURPOSE: ADDS DELIVERABLES FOR A PROJECT.
=====)

```

TABLE USED: DELIVER FORM USED: DELIVER

```
=====
SET VAR AGAIN = Y
SET NULL
WHILE AGAIN = Y THEN
    NEWPAGE
    WRITE "ENTER TO ADD DELIVERABLE FOR THIS PROJECT      ESC TO QUIT" AT 1 1
    DRAW DELIVER WITH ALL
    ENTER VAR VDEL RETURN ENTER ESC
    IF #RETURN = ESC THEN
        BREAK
    ENDIF
    LOAD DELIVER .VMID .VDEL
END
CLEAR VDEL
ENDWHILE
SET NULL -O-
RETURN
```

```
DMENU00
COLUMN DATABASE MANAGEMENT MAIN MENU
UNLOAD DATA TO SEND TO HOME
MERGE DATA FROM DISTRIBUTED SITE
PACK DATABASE TO RECOVER UNUSED SPACE
ARCHIVE RECORDS
REMOVE ALL ARCHIVED RECORDS TO HISTORY
RETURN TO MCASMIS MAIN MENU
```

```
DMENU04
COLUMN ARCHIVAL OF RECORDS MENU
ARCHIVE DOCUMENTS
ARCHIVE PROJECTS
RETURN TO DATABASE MANAGEMENT MAIN MENU
RETURN TO MCASMIS MAIN MENU
```

```
*( =====
PROGRAM: FIXDATE.CMD
AUTHOR : R.C.CAVALLARO
DATE WRITTEN: 29 NOVEMBER 1985
PURPOSE: THIS PROGRAM PROMPTS THE USER TO INPUT HOW MANY DAYS IN THE FUTURE SHOULD BE SEARCHED IN LOOKING FOR OVERDUE DOCUMENTS, PROJECTS OR UPCOMING DEADLINES. IT ADDS THIS NUMBER OF DAYS TO THE CURRENT DATE TO SET A TARGET DATE FOR THE SUBSEQUENT SEARCH.
```

```

=====
SET VAR DAYS INTEGER
SET VAR TGT DATE
SET VAR VALID = FALSE
SET ERROR VARIABLE ERV
WHILE VALID = FALSE THEN
  FILLIN DAYS USING "FOR HOW MANY DAYS IN THE FUTURE DO YOU WISH TO SEARCH:
  "
  IF ERV NE 480 THEN
    SET VAR VALID = TRUE
  ELSE
    WRITE "INVALID INPUT . . AN INTEGER IS REQUIRED"
    WRITE "INVALID INPUT . . AN INTEGER IS REQUIRED"
  ENDIF
ENDWHILE
SET VAR TGT = #DATE + . DAYS
NEWPAGE
CLEAR VALID
RETURN

```

This menu allows the user to choose how the output is presented for the subsequent queries and management reports. When the printer option is selected, output is routed to the screen also so that the paper may be repositioned by the user at the top of each page.

NOTE that some queries may not be routed to the printer directly. See the help screens for those queries or consult the users manual.

PRESS ANY KEY TO CONTINUE . . .

IOMENU  
ROW SELECT OUTPUT MENU  
SCREEN  
PRINTER

```

*( ===== PROGRAM: IOPICK.CMD
      AUTHOR : R. C. CAVALLARO
      DATE WRITTEN: 17 NOVEMBER 1985
      PURPOSE: DISPLAYS A MENU TO DETERMINE WHETHER THE USER DESIRES REPORT
                OUTPUT TO SCREEN OR DISK (OR ANY COMBINATION THEREOF).
                RETURNS OUTPUT TO SCREEN AT END OF REPORT.

SCREENS DISPLAYED: IOHLP, IOMENU
===== )

```

```

SET VAR PICK1 = GO
WHILE PICK1 = GO THEN
NEWPAGE
WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROWS AND ENTER"
WRITE "F10 FOR HELP AT 1 66
WRITE
CHOOSE PICK1 FROM IOMENU

IF PICK1 = HELP THEN
DISPLAY IOHLP
PAUSE
SET VAR PICK1 = GO
ENDIF

IF PICK1 = SCREEN THEN
SET VAR IO = N
OUTPUT SCREEN
ENDIF

IF PICK1 = PRINTER THEN
SET VAR IO = Y
ENDIF

ENDWHILE
RETURN

```

```

*( =====
PROGRAM: MAIN.CMD
AUTHOR : R. CAVALLARO
DATE WRITTEN: 17 NOVEMBER 1985
PURPOSE: THE MASTER MENU OF MCASMSI ROUTES CONTROL TO THE PROPER
MENU BASED ON THE USER'S RESPONSE.

PROGRAMS CALLED: SOOPICK.CMD, POOPICK.CMD, DBAPICK.CMD
SCREENS DISPLAYED: MAINHLP, MAINMENU
=====)

SET VAR CHOICE = 0          *{ initialize loop control variable)
SET MESSAGES OFF           *{ disable system messages )
SET ERROR MESSAGES OFF     *{ disable error messages )
SET ESCAPE OFF              *{ disable escape key to exit )
SET ERROR VAR ERV           *{ set error variable for trapping errors )
SET LINES 20                 *{ set terminal display to 20 lines/page )
SET DATE DDMMMYYY           *{ format date to military standard )
OPEN MCMSI                  *{ open the database )
WHILE CHOICE = 0 THEN      *{ print banner and main menu )
NEWPAGE
WRITE
WELCOME TO THE MARINE CORPS AUTOMATED SYSTEMS"

```

```

WRITE "
WRITE "
WRITE "
WRITE "YOU MAY PRESS F10 FOR HELP WHEN AT ANY OF THE MENU SCREENS"
WRITE "SELECT ANY OPTION WITH NUMBER OR CURSOR ARROW AND ENTER"
WRITE "CHOOSE "CHOICE FROM MAINMENU
WRITE "INPUT ACCEPTED . :*( PLEASE WAIT."
IF CHOICE = -1 THEN
  DISPLAY MAINHLP
  WRITE "PRESS ANY KEY TO CONTINUE . . ."
  PAUSE
  SET VAR CHOICE = 0
ENDIF

IF CHOICE = 1 THEN
  RUN SOOPICK.CMD
  *( systems management main menu )
ENDIF

IF CHOICE = 2 THEN
  RUN POOPICK.CMD
  *( project mgmt main menu )
ENDIF

IF CHOICE = 3 THEN
  RUN DOOPICK.CMD
ENDIF

IF CHOICE = 4 THEN
  BREAK
  *( return to operating system )
ENDIF

SET VAR CHOICE = 0
*( reset loop control variable )

ENDWHILE
CLEAR ALL VAR
SET MESSAGES ON
SET ERROR MESSAGES ON
SET ESCAPE ON
NEWPAGE
CLOSE
EXIT
RETURN

```

Select option 1 for Systems Management Processing including information about a system as a whole, about a development activity or about a particular document in the life cycle document package.

Select option 2 for Project Management functions to include opening and closing projects, establishing priorities or querying for analyst, type transaction code (TTC) changes and deliverables.

Select option 3 for Database Management Functions including archiving records, backing up files, packing the database to recover space, etc. This option is restricted to the Database Administrator.

Select option 4 to exit the MCASMS System and return to the operating system.

```
MAINMENU
COLUMN MCASMS MAIN MENU
SYSTEMS MANAGEMENT
PROJECT MANAGEMENT
DATABASE MANAGEMENT
RETURN TO OPERATING SYSTEM
```

```
*( =====
PROGRAM: MCASMS.BAT
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 9 FEBRUARY 1986
PURPOSE: THIS BATCH PROGRAM AUTOMATICALLY SETS THE PATH TO THE
          DIRECTORY CONTAINING THE R:BASE 5000 PROGRAMS AND INVOKES
          THE RBASE PROGRAM USING RBASE.DAT. THE DIRECTORY
          CONTAINING THE R:BASE PROGRAMS MUST RESIDE IN A DIRECTORY
          CALLED RBASE.
```

```
PROGRAMS CALLED: RBASE
```

```
=====
PATH RBASE
RBASE
EXIT
```

option 1 will add a new project, add one or more TICs to an existing project, or add a deliverable to an existing project. Option 2 provides the capability to change the data pertaining to a project, or a deliverable for a project. When the priority is changed for a project, all other priorities for that activity are automatically readjusted. Option 3 allows deletion of a project, TIC within a project, all projects for a system or a deliverable. When a project is deleted, all TIC and deliverable records pertaining to that project are automatically deleted. Option 4 provides different retrievals from the database by various selection criteria. Option 5 provides management reports pertaining to the project management functions.

```
*( =====
PROGRAM: POOPICK.CMD
AUTHOR : R. C. CAVALLARO
```

DATE WRITTEN: 17 NOVEMBER 1985  
PURPOSE: THE MAIN MENU OF THE PROJECT MANAGEMENT SUBSYSTEM ROUTES  
USER TO PROPER MENU BASED ON INDICATED INPUT TO MENU.

PROGRAMS CALLED: P01PICK.CMD, P02PICK.CMD, P03PICK.CMD, P04PICK.CMD,  
P05PICK.CMD  
SCREENS DISPLAYED: POOHLFP, PMENU00

```
=====
*(initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
    NEWPAGE
    WRITE "SELECT ANY OPTION WITH NUMBER OR CURSOR ARROW AND ENTER"
    WRITE "F10 FOR HELP AT 1 66"
    WRITE "H"
    CHOOSE "CHOICE FROM PMENU00"*(project mgmt main menu)
    WRITE "INPUT ACCEPTED . . PLEASE WAIT."
    IF CHOICE = -1 THEN
        DISPLAY POOHLFP
        WRITE "PRESS ANY KEY TO CONTINUE . . . "
        PAUSE
        SET VAR CHOICE = 0
    ENDIF
    IF CHOICE = 1 THEN
        RUN P01PICK.CMD
    ENDIF
    IF CHOICE = 2 THEN
        RUN P02PICK.CMD
    ENDIF
    IF CHOICE = 3 THEN
        RUN P03PICK.CMD
    ENDIF
    IF CHOICE = 4 THEN
        RUN P04PICK.CMD
    ENDIF
    IF CHOICE = 5 THEN
        RUN P05PICK.CMD
    ENDIF
    IF CHOICE = 6 THEN
        BREAK
    ENDIF
    =====
*
```

```
SET VAR CHOICE = 0
ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0
RETURN
```

\*( reset loop control variable)

Option 1 will provide a form for input of appropriate fields in the project record. After completing the form the user should press PGDN to load the data or Escape to quit without loading. ENTER tabs to next input area. Option 2 will provide an opportunity to add a TTC to an existing project. The user may continue to add TRCs by pressing ENTER, or ESC to quit. Option 3 provides a form for adding a deliverable to an existing project. The ENTER and ESC keys function the same as in option 2.

```
*( ====== PROGRAM: PO1PICK.CMD
      AUTHOR: R. C. CAVALLARO
      DATE WRITTEN: 7 JANUARY 1986
      PURPOSE: PROJECT MANAGEMENT ADD MENU. DISPLAYS PROPER FORM FOR
                ADDING A PROJECT OR ADDING ONE OR MORE TTCS TO AN EXISTING
                PROJECT. ERROR CHECKING IS ENFORCED BY
                THE RULES EXISTING IN THE DATA DICTIONARY.

      PROGRAMS CALLED: MAIN.CMD, P11PICK.CMD, P21PICK.CMD, P31PICK.CMD
      SCREENS DISPLAYED: PO1HLP, PMENU01
```

```
*( initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
  NEWPAGE
  WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
  WRITE "F10 FOR HELP" AT 1 66
  WRITE "INPUT ACCEPTED . . PLEASE WAIT."
  CHOOSE "CHOICE FROM PMENU01" *from Project mgmt add menu
  IF CHOICE = -1 THEN
    DISPLAY PO1HLP
    WRITE "PRESS ANY KEY TO CONTINUE . . . "
    PAUSE
  ENDIF
  IF CHOICE = 1 THEN
    RUN P11PICK.CMD
  ENDIF
```

```

IF CHOICE = 2 THEN          *( add TTC to existing project )
  RUN P21PICK.CMD
ENDIF

IF CHOICE = 3 THEN          *( add deliverable to existing project )
  RUN P31PICK.CMD
ENDIF

IF CHOICE = 4 THEN          *( return to project mgmt main menu )
  BREAK
ENDIF

IF CHOICE = 5 THEN          *( return to MCASMSIS main menu )
  QUIT TO MAIN.CMD
ENDIF

SET VAR CHOICE = 0          *( redisplay add menu )
ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0 *(reset loop control variable)
RETURN

```

Option 1: after changing the requested project, press PGDN to load, ENTER to tab to next input area, or ESC to quit without changing. If a priority for a project is changed, all other priorities for that activity are also automatically adjusted. Option 2: the user is prompted for an MPI ID, and then presented a form for all deliverables for that project (one at a time). For each deliverable shown, press PGDN to load changes, PGUP to find the next deliverable, or ESC to quit without changing.

```

*( =====
  PROGRAM: PO2PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 30 DECEMBER 1985
  PURPOSE: PROJECT MANAGEMENT EDIT MENU. DISPLAYS PROPER FORM FOR
            EDITING A PROJECT OR A DELIVERABLE FOR A PROJECT.

  PROGRAM CALLED: P12PICK.CMD, P22PICK.CMD, MAIN.CMD
  SCREENS DISPLAYED: PMENU02, PO2HELP
  =====)

*( initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
  NEWPAGE
  SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
  WRITE

```

```

WRITE "F10 FOR HELP" AT 1 66
WRITE "CHOOSE CHOICE FROM PMENU02"
WRITE "INPUT ACCEPTED . . PLEASE WAIT."
WRITE "PAUSE"

IF CHOICE = -1 THEN *(F10 pressed for help)
DISPLAY PO2HLP
WRITE "PRESS ANY KEY TO CONTINUE . . ."
PAUSE
ENDIF

IF CHOICE = 1 THEN *(edit the project )
RUN P12PICK.CMD
ENDIF

IF CHOICE = 2 THEN *( edit a deliverable )
RUN P22PICK.CMD
ENDIF

IF CHOICE = 3 THEN *( return to systems management main menu )
BREAK
ENDIF

IF CHOICE = 4 THEN *( quit to MCASMSI main menu )
QUIT TO MAIN.CMD
ENDIF

SET VAR CHOICE = 0 *( redisplay edit menu )
ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0 *( reset loop control variable )
RETURN

```

Option 1 deletes a single project.  
 Option 2 deletes all projects for a system.  
 Option 3 deletes one or more TTCS from a project.  
 Option 4 deletes a deliverable from a project.

In all options, the user is first presented the information in the appropriate form to verify that the record should be deleted. Press PGDN to delete the record shown, PGUP to find the next match, or ESC to quit and return to the last menu. The process of deleting a record requires confirmation and the user is given one additional chance to quit without changing in case a wrong key was pressed. When a project is deleted, all priorities for that activity are automatically adjusted.

```

*( =====
PROGRAM: P03PICK.CMD
AUTHOR: R. C. CAVALLARO
DATE WRITTEN: 30 DECEMBER 1985
PURPOSE: PROJECT MANAGEMENT DELETE MENU. DELETES PROJECTS FOR AN ENTIRE
SYSTEM OR BY MP IDENTIFICATION NUMBER. PROJECTS FOR A PROJECT
OR DELIVERABLES FOR A PROJECT. WHEN A PROJECT IS DELETED ALL
DELIVERABLES FOR THAT PROJECT ARE AUTOMATICALLY DELETED ALSO.

PROGRAMS CALLED: MAIN.CMD P13PICK.CMD, P23PICK.CMD, P33PICK.CMD,
P43PICK.CMD
SCREENS DISPLAYED: P03HLP, PMENU03
=====)

SET VAR CHOICE = 0      *( initialize loop control variable )
WHILE CHOICE = 0 THEN
NEWPAGE
WRITE " SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER "
WRITE " F10 FOR HELP AT 1 66 "
WRITE " "
CHOOSE " CHOICE FROM PMENU03 "
WRITE " INPUT ACCEPTED . . PLEASE WAIT. "
WRITE " "

IF CHOICE = -1 THEN    *( F10 pressed for help )
DISPLAY P03HLP
WRITE " PRESS ANY KEY TO CONTINUE . . . "
PAUSE
ENDIF

IF CHOICE = 1 THEN     *( delete a single project )
RUN P13PICK.CMD
ENDIF

IF CHOICE = 2 THEN     *( delete all projects for a system )
RUN P23PICK.CMD
ENDIF

IF CHOICE = 3 THEN     *( delete one or more TTC's from a project )
RUN P33PICK.CMD
ENDIF

IF CHOICE = 4 THEN     *( delete deliverable from a project )
RUN P43PICK.CMD
ENDIF

IF CHOICE = 5 THEN     *( return to systems management main menu )
BREAK
ENDIF

```

```

IF CHOICE = 6 THEN          *( quit to MCASMS main menu )
QUIT TO MAIN.CMD
ENDIF

SET VAR CHOICE = 0           *( redisplay delete menu )

ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0           *( reset loop control variable )
RETURN

```

The user is provided with selected information based on the option selected and the answers to the subsequent prompts. See the users manual for more detailed instructions concerning each prompt. Output is routed to the device last designated using option 1 of this menu except for options 6 and 7 which cannot go directly to the printer.

```

*( =====
PROGRAM: PO4PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: OUTPUTS SELECTED INFORMATION DESIRED BY USER AFTER DISPLAYING
          SELECTION CRITERIA

PROGRAM CALLED: PROJDRAW. CMD IOPICK. CMD, MAIN. CMD
SCREENS DISPLAYED: PO4HLP, PMENU04
TABLES USED: PROJECT, TTC, DELIVER
*( =====

SET NULL " "                  *( display null values as blanks )
SET VAR IO = N
SET VAR CHOICE = 0             *( initialize loop control variable )
SET ERROR VAR ERV              *( set error variable for trapping )
WHILE CHOICE = 0 THEN
NEWPAGE
WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
WRITE "F10 FOR HELP AT 1 66
CHOOSE "CHOICE FROM PMENU04
WRITE "INPUT ACCEPTED . . PLEASE WAIT."
WRITE

IF CHOICE = -1 THEN          *( F10 pressed for help )
DISPLAY "PO4HLP
WRITE "PRESS ANY KEY TO CONTINUE . . .
PAUSE
SET VAR CHOICE = 0

```

```

ENDIF
      *( change output device )
      IF CHOICE = 1 THEN
        RUN TOPICK.CMD
      ENDIF

      IF CHOICE = 2 THEN
        NEWPAGE
        IF IO = Y THEN
          OUTPUT SCREEN WITH PRINTER
          SET LINES 56
        ENDIF
        SELECT MPI-ID SYSNAME PRIORITY=4 DEVACT PROJNAME=20 SCHEDDUE +
              FROM PROJECT SORTED BY SYSNAME PRIORITY WHERE STATUS = O
        OUTPUT SCREEN
        SET LINES 20
      ENDIF

      IF CHOICE = 3 THEN
        NEWPAGE
        IF IO = Y THEN
          OUTPUT SCREEN WITH PRINTER
          SET LINES 56
        ENDIF
        SELECT MPI-ID SYSNAME DEVACT PROJNAME=20 SCHEDDUE CAT FROM PROJECT +
              SORT BY SCHEDDUE WHERE STATUS = O
        OUTPUT SCREEN
        SET LINES 20
      ENDIF

      IF CHOICE = 4 THEN
        NEWPAGE
        IF IO = Y THEN
          OUTPUT SCREEN WITH PRINTER
          SET LINES 56
        ENDIF
        SELECT MPI-ID SYSNAME DEVACT PROJNAME=20 SCHEDDUE CAT FROM PROJECT +
              SORT BY SCHEDDUE WHERE STATUS = O
        OUTPUT SCREEN
        SET LINES 20
      ENDIF

      IF CHOICE = 5 THEN
        WRITE "ENTER ACTIVITY ABBREVIATION OR ENTER L FOR A LIST OF ALL
              FILLIN CH1 USING CURRENT ACTIVITY
        NEWPAGE
        IF IO = Y THEN
          OUTPUT SCREEN WITH PRINTER
          SET LINES 56
        ENDIF
        IF CH1 = L THEN
          SELECT DEVACT DEVLONG FROM VENDOR SORT BY DEVACT
        ELSE
          SELECT MPI-ID SYSNAME PROJNAME=30 SCHEDDUE CAT=3 PRIORITY=4 FROM +
                PROJECT SORTED BY PRIORITY WHERE DEVACT = .CH1 AND STATUS = O
        ENDIF
        OUTPUT SCREEN
        SET LINES 20
      ENDIF
    
```

```

ENDIF
IF CHOICE = 6 THEN
  RUN TTCXREF.CMD
ENDIF

IF CHOICE = 7 THEN
  WRITE "ENTER THE MPI PROJECT ID OR ENTER # FOR A LIST OF ALL CURRENT"
  FILLIN CH1 USING
  IF CH1 = L THEN
    NEWPAGE
    SELECT MPI-ID SYSNAME PROJNAME=54 FROM PROJECT SORT BY MPI-ID
  ELSE
    SET POINTER #1 PT1 FOR PROJECT WHERE MPI-ID = .CH1
    IF PT1 NE 0 THEN
      WRITE "THERE IS NO PROJECT ON FILE BY THAT NAME"
    ELSE
      NEWPAGE
      SET NULL -O- *( default null value needed for dates )
      RUN PROJDRAW.CMD *( display null as blanks in this program )
      SET NULL "
      WRITE "
      WRITE "PRESS ANY KEY TO CONTINUE . "
      PAUSE
      NEWPAGE
      SELECT "TTC FROM TTC SORT BY TTC WHERE MPI-ID = .CH1
      WRITE "
      SELECT DELIVER FROM DELIVER WHERE MPI-ID = .CH1
      ENDIF
    ENDIF
  ENDIF
ENDIF

IF CHOICE = 8 THEN
  BREAK
ENDIF

IF CHOICE = 9 THEN
  QUIT TO MAIN.CMD
ENDIF

IF CHOICE NE 0 AND CHOICE NE 1 THEN
  WRITE "
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ENDIF
SET VAR CHOICE = 0
ENDWHILE
CLEAR ALL VAR
OUTPUT SCREEN
SET VAR CHOICE = 0
*( reset loop control variable )

```

```
SET NULL -0-
RETURN
```

```
*( reset default null indicator )
```

Option 1: changes output device until menu user returns to previous menu  
or rechanges output destination using this option again.  
Option 2: lists overdue or soon to be due project implementations.  
Option 3: lists open projects by functional or activity project officer.  
Option 4: log of projects by date of MPI request.  
Option 5: log of deliverables for all projects on file.  
Option 6: log of projects by scheduled due date.  
Option 7: Test cycle report including open and closed projects,  
indicating the amount of testing that has been conducted.

```
*( =====
PROGRAM: PO5PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 8 JANUARY 1986
PURPOSE: OUTPUTS SELECTED REPORTS TO THE DESIGNATED OUTPUT DEVICE BASED
ON USER SELECTION.

PROGRAM CALLED: IOPICK.CMD MAIN.CMD, FIXDATE.CMD
SCREENS DISPLAYED: PO5HLP PMENU05
TABLES USED: PROJECT, TTC, DELIVER
=====)

SET NULL " " *( display null values as blanks )
SET VAR IO = N
SET VAR CHOICE = 0 *( initialize loop control variable )
SET ERROR VAR ERV *( set error variable for trapping )
WHILE CHOICE = 0 THEN
  NEWPAGE
  WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
  WRITE "F10 FOR HELP AT 1 66"
  CHOOSE "CHOICE FROM PMENU05"
  WRITE "INPUT ACCEPTED . PLEASE WAIT."
  WRITE

IF CHOICE = -1 THEN
  DISPLAY PO5HLP
  WRITE "PRESS ANY KEY TO CONTINUE . . . "
  PAUSE
  SET VAR CHOICE = 0
ENDIF

IF CHOICE = 1 THEN
  RUN IOPICK.CMD
ENDIF
```

```

IF CHOICE = 2 THEN          *( overdue/soon due projects )
  RUN FIXDATE.CMD
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
  ENDIF
  PRINT PROJDUE SORTED BY SYSNAME SCHEDDUE WHERE STATUS = O AND +
    SCHEDDUE LE .TGT
  OUTPUT SCREEN
ENDIF

IF CHOICE = 3 THEN          *( projects by func/activity PO )
  SET VAR ANS = C
  WHILE ANS NE F AND ANS NE A THEN
    NEWPAGE
    WRITE "F - BY FUNCTIONAL ANALYST"
    WRITE "A - BY ACTIVITY PROJECT OFFICER"
    WRITE " "
    FILLIN ANS USING "ENTER EITHER F OR A AS DESIRED: "
  ENDWHILE
  IF ANS = F THEN
    SET VAR PERSON = "FUNCPO"
  ELSE
    SET VAR PERSON = "ACTIVPO"
  ENDIF
  WRITE " "
  WRITE "ENTER THE PERSON'S NAME OR ENTER L FOR A LIST OF ALL NAMES"
  FILLIN CH1 USING "ON FILE FOR THIS FIELD:
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  IF CH1 = L THEN
    TALLY .PERSON IN PROJECT
  ELSE
    SELECT MPI-ID SYSNAME PRIORITY=4 DEVACT PROJNAME=20 SCHEDDUE FROM +
      PROJECT SORTED BY MPI-ID WHERE .PERSON = .CH1
  ENDIF
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF CHOICE = 4 THEN          *( log of all projects by date requested )
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
  ENDIF
  PRINT PROJLOG SORTED BY DATEREQ=D
  OUTPUT SCREEN

```

```

ENDIF
IF CHOICE = 5 THEN      *( log of deliverables for projects )
  IF LO = Y THEN
    OUTPUT SCREEN WITH PRINTER
  ENDIF
  PRINT DELLOG SORTED BY MPI-ID
  OUTPUT SCREEN
ENDIF

IF CHOICE = 6 THEN      *( log of all project by scheduled due date )
  NEWPAGE
  IF LO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT MPI-ID PROJNAME DEVACT=9 PROJNAME=24 SCHEDDUE STATUS=4 +
  FROM PROJECT SORTED BY SCHEDDUE
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF CHOICE = 7 THEN      *( test cycle report )
  SET VAR CH1 DATE
  FILLIN CH1 USING "ENTER THE TEST CYCLE IMPLEMENTATION DATE: "
  FILLIN CH2 USING "ENTER THE SYSTEM ABBREVIATION:
  NEWPAGE
  IF LO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT PROJ-ID MPI-ID PROJNAME=33 TESTCOND=8 TESTED=6 STATUS FROM +
  PROJECT SORTED BY PROJ-ID WHERE SCHEDDUE = .CH1 AND SYSNAME = .CH2
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF CHOICE = 8 THEN      *( return to project management main menu )
  BREAK
ENDIF

IF CHOICE = 9 THEN      *( return to MCASMIS main menu )
  QUIT TO MAIN.CMD
ENDIF

IF CHOICE NE 0 AND CHOICE NE 1 THEN
  WRITE "PRESS ANY KEY TO CONTINUE . . ." *( pause )
  PAUSE

```

```

ENDIF
SET VAR CHOICE = 0          *(* redisplay query menu )
ENDWHILE
OUTPUT SCREEN
CLEAR ALL VAR
SET VAR CHOICE = 0          *(* reset loop control variable )
SET NULL -0-
RETURN

```

```

*( =====
PROGRAM: P11PICK.CMD
AUTHOR : R. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: ADD A NEW PROJECT. AUTOMATICALLY ADJUSTS PRIORITIES OF
OTHER PROJECTS FOR THAT DEVELOPMENT ACTIVITY IF NECESSARY.
PRIORITY MAY NOT BE ASSIGNED A HIGHER NUMBER THAN THE
NUMBER OF PROJECTS ON FILE FOR THAT ACTIVITY. THIS PROGRAM
AUTOMATICALLY TAKES CARE OF THIS FACT.

```

```

PROGRAMS CALLED: PROJLOAD.CMD, ADDTTC.CMD, DELIVADD.CMD
TABLES USED: PROJECT
VARIABLE FORMS USED: PROJECT

```

```

=====
SET VAR MORE TO "Y"          *(* initialize loop control variable)
WHILE MORE = Y THEN
NEWPAGE
DRAW PROJECT
WRITE "ENTER TO TAB TO NEXT FIELD" + *( variable form project with null values)
AT 1 TO LOAD ESC TO QUIT" +
*( user inputs values in form and continues after pressing
ENTER VAR VMINID VALID VSYS VPRI VSTAT VCAT VPROJ VPO VAPO VDEV +
VSWT VREQ VDUE VSCHED VCOMP VPID VTC VTEST +
RETURN PGDN ESC
#RETURN = ESC THEN *( if user pressed ESC then quit)
BREAK
ENDIF
*( calculate the highest priority for that activity
COMPUTE HIPRI AS MAX_PRIORITY FROM PROJECT WHERE DEVACT = .VDEV +
AND STATUS = O
IF HIPRI FAILS THEN *( no open projects for this activity )
SET VAR HIPRI = 0
ENDIF
*( change priority numbers if applicable
IF VPRI EXISTS THEN
IF VPRI > HIPRI THEN *{ if priority input > highest yet }
SET VAR VPRI TO .HIPRI + 1
IF VPRI > HIPRI THEN *{ if priority allow too high priority }

```

```

ELSEASSIGN PRIORITY TO PRIORITY + 1 IN PROJECT WHERE PRIORITY +
      GE .VPRI AND DEVACT = .VDEV
ENDIF
ELSESET VAR VPRI TO .HIPRI + 1 *( assign lowest priority )
ENDIF
*(load the new record into the project table)
RUN PROJLOAD.CMD
IF OKAY = TRUE THEN
  IF VSYS = JUMPS/MMS OR VSYS *( if load was successful )
    RUN ADDTC.CMD
  ENDIF
  RUN DELIVADD.CMD
ENDIF
CLEAR ALL VAR
SET VAR CHOICE = 0
*( prompt for another project addition {Y/N): " AT 11 25
FILLIN MORE USING ADD ANOTHER PROJECT
ENDWHILE
RETURN

*( ====== PROGRAM: P12PICK.CMD
  AUTHOR : R. C. CAVALLARO
  DATE WRITTEN: 7 JANUARY 1986
  PURPOSE: EDIT A PROJECT. AUTOMATICALLY ADJUSTS PRIORITIES OF
            OTHER PROJECTS FOR THAT DEVELOPMENT ACTIVITY IF NECESSARY.
            PRIORITY MAY NOT BE ASSIGNED A HIGHER NUMBER THAN THE
            NUMBER OF PROJECTS ON FILE FOR THAT ACTIVITY. THIS PROGRAM
            AUTOMATICALLY TAKES CARE OF THIS FACT.

PROGRAMS CALLED: PROJLOAD.CMD, PROJDRAW.CMD
TABLES USED: PROJECT
=====)

FILLIN CH1 USING "ENTER THE MPI PROJECT IDENTIFICATION: "
SET POINTER #1 PT1 FOR PROJECT WHERE MPI-ID = .CH1
IF PT1 NE "0" THEN
  *( matching record not found ) ADD IT"
  WRITE "THIS PROJECT IS NOT IN THE FILE . YOU MUST ADD IT"
  WRITE "PRESS ANY KEY TO CONTINUE . . "
PAUSE
ELSE
  NEWPAGE
  RUN PROJDRAW.CMD *( variable form with values from record )
  SET VAR CHECK = VPRI *( save old priority )
  WRITE "ENTER TO TAB PGDN TO LOAD

```

```

AT 1 1 VALID VSYS VPRI VSTAT VCAT VPROJ VPO VDEV +
VSWT VREQ VDUE VSCHED VCOMP VPID VTEST RETURN PGDN ESC
IF #RETURN = PGDN THEN
  IF *( check if priorities need to be reset
    COMPUTE HIPRI AS MAX PRIORITY FROM PROJECT WHERE DEVACT = .VDEV +
    AND STATUS = O
    IF VPRI GT HIPRI THEN
      SET VAR VPRI TO .HIPRI + 1
    ENDIF
    IF VPRI GT CHECK THEN
      ASSIGN PRIORITY TO PRIORITY - 1 IN PROJECT WHERE PRIORITY +
      LE .VPRI AND PRIORITY GT .CHECK AND DEVACT = .DEV
    ENDIF
    IF VPRI LT CHECK THEN
      ASSIGN PRIORITY TO PRIORITY + 1 IN PROJECT WHERE PRIORITY +
      GE .VPRI AND PRIORITY LT .CHECK AND DEVACT = .DEV
    ENDIF
    *( save the change if PGDN )
    RUN PROJLOAD.CMD *{ load the new row
    IF OKAY = TRUE THEN *{ if load was successful
      DELETE ROWS FROM #1 *( delete the old row )
    ENDIF
  ENDIF
  RETURN

```

\*( ======  
 PROGRAM: P13PICK.CMD  
 AUTHOR : R. CAVALLARO  
 DATE WRITTEN: 7 JANUARY 1986  
 PURPOSE: DELETE A SINGLE PROJECT. AUTOMATICALLY ADJUSTS THE  
 PRIORITIES OF ALL OTHER PROJECTS FOR THAT DEVELOPMENT  
 ACTIVITY. ALSO DELETES TTC AND DELIVERABLE RECORDS  
 CONNECTED WITH THAT DELETED PROJECT.

PROGRAMS CALLED: PROJDRAW.CMD  
 TABLES USED: PROJECT, TTC, DELIVER

=====)  
 FILLIN CH1 USING "WHICH MPI IDENTIFICATION NUMBER TO DELETE: "
 SET POINTER #1 PT1 FOR PROJECT WHERE MPI-ID = .CH1
 IF PT1 NE O THEN \*( no such project )
 WRITE "THIS PROJECT IS NOT IN THE FILE - IT CANNOT BE DELETED"
 WRITE "PRESS ANY KEY TO CONTINUE . . "
 PAUSE
 ELSE

```

NEWPAGE
RUN PROJDRAW.CMD *{ draw project" form with values of matching record )
WRITE "PGDN TO DELETE THIS RECORD" AT 11
WRITE "ESC TO QUIT AT 160
EDIT VAR VMID RETURN PGDN ESC
IF #RETURN = PGDN THEN
  NEWPAGE
  WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
  FILLIN ANS USING RECORD *{ Y/N
  IF ANS = Y THEN
    *{ reset the priorities
      ASSIGN PRIORITY TO #1 IN PROJECT WHERE PRIORITY +
      GT VPRI AND DEVACT = .VDEV
      DELETE ROWS FROM #1
      DELETE ROWS FROM TTC WHERE MPI-ID = .CH1
      DELETE ROWS FROM DELIVER WHERE MPI-ID = .CH1
    ENDIF
  ENDIF
  RETURN

```

```

* ( =====
  PROGRAM: P21PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 7 JANUARY 1986
  PURPOSE: ADD A TTC TO AN EXISTING PROJECT. THIS PROGRAM IS USED
           FOR JUMPS/MMS AND REMMPS TO TRACK THE TYPE TRANSACTION
           CODES THAT ARE MODIFIED, CREATED OR DELETED BY A GIVEN
           PROJECT.

PROGRAMS CALLED: ADDTTC. CMD
TABLES USED: PROJECT
=====)

```

64

```

SET VAR MORE TO "Y"
WHILE MORE = Y THEN
  NEWPAGE
  WRITE "ENTER MPI PROJECT ID OR ENTER L" FOR A LIST OF ALL CURRENT"
  FILLIN VMID USING "MPI-ID" ON FILE: AT 107
  IF VMID = L THEN
    SELECT MPI-ID SYSNAME PROJNAME=54 FROM PROJECT SORTED BY MPI-ID
    SET POINTER #1 PT1 FOR PROJECT WHERE MPI-ID = .VMID
    IF PT1 = 0 THEN
      NEWPAGE
      RUN ADDTTC. CMD
    ELSE
      WRITE "NO RECORD ON FILE FOR THIS MPI PROJECT ID" AT 12 4

```

```

ENDIF
CLEAR ALL VAR
SET VAR CHOICE = 0
FILLIN MORE USING "ADD TTC TO ANOTHER PROJECT (Y/N)? " AT 22 5
ENDWHILE
RETURN

```

```

* =====
* PROGRAM: P22PICK.CMD
* AUTHOR : R. C. CAVALLARO
* DATE WRITTEN: 7 JANUARY 1986
* PURPOSE: EDIT A DELIVERABLE CYCLES THROUGH ALL DELIVERABLES
*           FOR A GIVEN PROJECT, ALLOWING CHANGES TO BE MADE TO
*           DESIRED RECORDS.

```

TABLES USED: DELIVER  
VARIABLE FORMS USED: DELIVER

```

===== )
=====

65 FILLIN CH1 USING "ENTER THE MPI PROJECT IDENTIFICATION:
SET POINTER #1 PT1 FOR DELIVER WHERE MPI-ID = CH1
IF PT1 NE 0 THEN *(
SET VAR VDEL TO DELIVER IN #1
SET VAR VMID TO MPI-ID IN #1
SET VAR VDEL TO DELIVER IN #1
NEWPAGE
SET NULL " "
DRAW DELIVER WITH ALL *(
SET NULL-O-
WRITE \#PGDN TO LOAD THE CHANGES
AT 1 1
WRITE \#PGUP FOR NEXT MATCH" AT 1 60
*( edit deliverable )
EDIT VAR VDEL RETURN ESC PGDN PGUP
IF #RETURN = ESC THEN *( quit if ESC was pressed )
BREAK
ENDIF
IF #RETURN = PGDN THEN *(
    save changes if PGDN pressed
) CHANGE DELIVER TO .VDEL IN #1
ENDIF
NEXT #1 PT1      *( find next matching record )

```

```
ENDWHILE  
ENDIF  
RETURN
```

```
* ( ======  
PROGRAM: P23PICK.CMD  
AUTHOR : R. C. CAVALLARO  
DATE WRITTEN: 7 JANUARY 1986  
PURPOSE: DELETE ALL PROJECTS FOR ONE SYSTEM. ADJUSTS THE  
PRIORITY OF ALL OTHER PROJECTS FOR THAT DEVELOPMENT  
ACTIVITY. ALSO DELETES TTC AND DELIVERABLE RECORDS  
CONNECTED WITH THOSE DELETED PROJECTS.
```

```
TABLES USED: PROJECT, TTC, DELIVER
```

```
===== )  
FILLIN CH1 USING "ENTER SYSTEM NAME:  
SET POINTER #1 PT1 FOR PROJECT WHERE SYSNAME = .CH1  
IF PT1 NE 0 THEN * ( no such projects )  
WRITE "THERE ARE NO PROJECTS ON FILE FOR THIS SYSTEM"  
WRITE "PLEASE CHECK AND RE-ENTER IF NECESSARY."  
WRITE "PRESS ANY KEY TO CONTINUE"  
PAUSE  
ELSE  
WRITE "  
FILLIN ANS USING "PROJECTS(Y/N)  
IF ANS = Y THEN * ( delete projects after one last chance )  
WHILE PT1 = 0 THEN  
SET VAR CH2 = MPI-ID IN #1  
SET VAR VSYS = SYSNAME IN #1  
SET VAR VDEV = DEVACT IN #1  
SET VAR VPRI = PRIORITY IN #1  
* ( reset the priorities )  
ASSIGN PRIORITY TO PRIORITY - 1 IN PROJECT WHERE +  
PRIORITY GT .VPRI AND DEVACT = .VDEV  
DELETE ROWS FROM TTC WHERE MPI-ID = .CH2  
DELETE ROWS FROM DELIVER WHERE MPI-ID = .CH2  
NEXT #1 PT1  
ENDWHILE  
DELETE ROWS FROM PROJECT WHERE SYSNAME = .CH1  
ENDIF  
ENDIF  
RETURN  
* ( ======
```

PROGRAM: P31PICK.CMD  
 AUTHOR : R. C. CAVALLARO  
 DATE WRITTEN: 7 JANUARY 1986  
 PURPOSE: ADD A DELIVERABLE TO AN EXISTING PROJECT.  
 PROGRAMS CALLED: DELIVADD.CMD  
 TABLES USED: PROJECT

```

=====
SET VAR MORE TO "Y"          * ( initialize loop control )
WHILE MORE = Y THEN
  NEWPAGE
  WRITE ENTER THE MPI PROJECT ID OR ENTER L FOR A LIST OF ALL CURRENT"
  FILLIN VMID USING MPI PROJECT ID'S : AT 10 7
  IF VMID = L THEN
    SELECT MPI-ID SYSNAME PROJNAME=54 FROM PROJECT SORTED BY MPI-ID
  ELSE
    SET POINTER #1 PT1 FOR PROJECT WHERE MPI-ID = .VMID
    IF PT1 = O THEN
      NEWPAGE
      RUN DELIVADD.CMD
    ELSE
      WRITE "NO RECORD ON FILE FOR THIS MPI PROJECT ID" AT 12 4
    ENDIF
  ENDIF
  CLEAR ALL VAR
  SET VAR CHOICE = 0
  FILLIN MORE USING "ADD DELIVERABLE TO ANOTHER PROJECT ( Y/N)? " AT 22 5
ENDWHILE
RETURN
=====
```

```

*( =====
PROGRAM: P33PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: DELETE ONE OR MORE TTC'S FROM A GIVEN PROJECT.
TABLES USED: TTC
VARIABLE FORMS USED: TTC
=====
```

```

FILLIN CH1 USING "ENTER MPI PROJECT IDENTIFICATION NUMBER: "
SET POINTER #1 PT1 FOR TTC WHERE MPI-ID = .CH1
IF PT1 NE O THEN
  * ( no such MPI ID )
  WRITE "THIS PROJECT IS NOT ON FILE - TTC's FOR "
  WRITE "IT CAN NOT BE DELETED"
=====
```

```

        WRITE "PRESS ANY KEY TO CONTINUE . . "
        PAUSE

        ELSE WHILE PT1 = 0 THEN
          SET VAR VMID TO MPI-ID IN #1
          SET VAR VTTC TO TTC IN #1
          NEWPAGE
          DRAW TTC WITH ALL
          WRITE PGDN TO DELETE THIS TTC
          WRITE "ESC TO QUIT" AT 1 60
          EDIT VAR VTTC RETURN PGDN ESC PGUP
          IF #RETURN = ESC THEN
            *( quit if ESC was pressed )
            ENDIF
          IF #RETURN = PGDN THEN
            *( delete the record )
            NEWPAGE
            WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
            FILLIN ANS USING #TTC (Y/N)
            IF ANS = Y THEN
              DELETE ROWS FROM #1 *( after one last chance )
            ENDIF
            ENDIF
            NEXT #1 PT1
          ENDWHILE
        ENDIF
        RETURN

        *(
        ====== PROGRAM: P43PICK.CMD
        AUTHOR : R. C. CAVALLARO
        DATE WRITTEN: 7 JANUARY 1986
        PURPOSE: DELETE ONE OR MORE DELIVERABLES FROM A PROJECT.
        TABLES USED: DELIVER
        VARIABLE FORMS USED: DELIVER
        ======)

        FILLIN CH1 USING "ENTER MPI PROJECT IDENTIFICATION NUMBER: "
        SET POINTER #1 PT1 FOR DELIVER WHERE MPI-ID = CH1
        IF PT1 NE 0 THEN
          *( no such project on file )
          WRITE ""
          WRITE "THERE ARE NO DELIVERABLES FOR THIS PROJECT ON FILE"
          WRITE "PRESS ANY KEY TO CONTINUE . . "
          PAUSE
        ELSE WHILE PT1 = 0 THEN
          SET VAR VMID TO MPI-ID IN #1

```

```

SET VAR VDEL TO DELIVER IN #1
NEWPAGE
DRAW DELIVER WITH ALL
WRITE "PGDN TO DELETE THIS DELIVERABLE      PGUP FOR NEXT MATCH" +
      AT 1 1
WRITE "ESC TO QUIT" AT 1 60
EDIT VAR VDEL RETURN ESC PGDN PGUP
IF #RETURN = ESC THEN
  BREAK
ENDIF
IF #RETURN = PGDN THEN
  *( delete the record )
  NEWPAGE
  WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
  FILL IN ANS USING "DELIVERABLE (Y/N)"
  IF ANS = Y THEN
    *( after one last chance )
    DELETE ROWS FROM #1
  ENDIF
ENDIF
NEXT #1 PT1
ENDWHILE
ENDIF
RETURN

```

```

PMENU00
COLUMN PROJECT MANAGEMENT MAIN MENU
ADD
CHANGE
DELETE
QUERY
MANAGEMENT REPORTS
RETURN TO MCASMS MAIN MENU

```

```

ENDWHILE
ENDIF
RETURN

```

```

PMENU01
COLUMN PROJECT MANAGEMENT ADD MENU
ADD A NEW PROJECT
ADD TTC TO EXISTING PROJECT
ADD DELIVERABLE TO EXISTING PROJECT
RETURN TO PROJECT MANAGEMENT MAIN MENU
RETURN TO MCASMS MAIN MENU

```

```

PMENU02
COLUMN PROJECT MANAGEMENT EDIT MENU
EDIT A PROJECT
EDIT A DELIVERABLE FOR A PROJECT
RETURN TO PROJECT MANAGEMENT MAIN MENU
RETURN TO MCASMS MAIN MENU

```

PMENU03  
COLUMN PROJECT MANAGEMENT DELETE MENU  
DELETE A SINGLE PROJECT  
DELETE ALL PROJECTS FOR ONE SYSTEM  
DELETE A TTC FOR A PROJECT  
DELETE A DELIVERABLE FOR A PROJECT  
RETURN TO PROJECT MANAGEMENT MAIN MENU  
RETURN TO MCASMS MAIN MENU

PMENU04  
COLUMN PROJECT MANAGEMENT QUERY MENU  
CHANGE OUTPUT DEVICE  
ALL OPEN PROJECTS SORTED BY SYSTEM  
ALL OPEN PROJECTS FOR ONE SYSTEM  
ALL OPEN PROJECTS SORTED BY SCHEDULED DUE DATE  
ALL OPEN PROJECTS FOR AN ACTIVITY BY PRIORITY  
TTC CROSS REFERENCE MENU  
ALL INFORMATION ON A PARTICULAR PROJECT  
RETURN TO PROJECT MANAGEMENT MAIN MENU  
RETURN TO MCASMS MAIN MENU

PMENU05  
COLUMN PROJECT MANAGEMENT MANAGEMENT REPORTS  
CHANGE OUTPUT DEVICE  
OPEN PROJECTS THAT ARE OVERDUE OR DUE SOON  
PROJECTS BY FUNCTIONAL OR ACTIVITY PROJECT OFFICER  
LOG OF PROJECTS BY DATE OF MPI REQUEST  
LOG OF DELIVERABLES FOR PROJECTS ON FILE  
LOG OF PROJECTS BY SCHEDULED DUE DATE  
TEST CYCLE STATUS REPORT  
RETURN TO PROJECT MANAGEMENT MAIN MENU  
RETURN TO MCASMS MAIN MENU

\* ( ======  
PROGRAM: PROJDRAW.CMD  
AUTHOR : R. C. CAVALLARO  
DATE WRITTEN: 20 DECEMBER 1985  
PURPOSE: SETS ALL OF THE VARIABLE CONTENTS BEFORE DRAWING THE  
FORM PROJECT FOR EDITING VARIOUS DATA.  
VARIABLE FORM USED: PROJECT  
===== )

SET VAR VMID TO MPI-ID IN #1  
SET VAR VAID TO ACTIV-ID IN #1  
SET VAR VPID TO PROJ-ID IN #1

```

SET VAR VSSYS TO SYSTYPE IN #1
SET VAR VCAT TO CAT IN #1 PRIORITY IN #1
SET VAR VPROJ TO FUNCPO IN #1
SET VAR VFPO TO ACTIVPO IN #1
SET VAR VAPO TO STATUS IN #1
SET VAR VSTAT TO DEACT IN #1
SET VAR VDEV TO PROJNAME IN #1
SET VAR VPROJ TO STRUCWT IN #1
SET VAR VSWT TO DATEREQ IN #1
SET VAR VREQ TO REQDUE IN #1
SET VAR VDUE TO SCHEDDUE IN #1
SET VAR VSCHED TO COMPLETE IN #1
SET VAR VCOMP TO TESTCOND IN #1
SET VAR VTC TO TESTED IN #1
SET VAR VTEST TO TESTED IN #1
SET NULL -O- DRAW PROJECT WITH ALL
SET NULL -O- RETURN

```

```

*( ====== PROGRAM: PROJLIST CMD
      AUTHOR : R. C. CAVALLARO
      DATE WRITTEN: 6 JANUARY 1986
      PURPOSE: THIS PROGRAM LISTS ALL OPEN PROJECTS FOR ANY SYSTEM AFTER
              PROMPTING THE USER TO INPUT A SYSTEM NAME.
===== )

```

TABLES USED: PROJECT

```

NEWPAGE
WRITE "ENTER THE SYSTEM ABBREVIATION OR ENTER L FOR A LIST OF ALL VALID"
FILLIN CH1 USING "SYSTEM ABBREVIATIONS ON FILE:
IF CH1 = L THEN
  SELECT SYSTYPE LNGTITLE=60 FROM SYSTEM SORTED BY SYSTYPE
ELSE
  SET POINTER #1 PT1 FOR PROJECT WHERE SYSTYPE = .CH1 AND STATUS = O
  IF PT1NE O THEN
    WRITE ""
    WRITE "THERE ARE NO OPEN PROJECTS ON FILE FOR THIS SYSTEM"
  ELSE
    NEWPAGE
    IF IO = Y THEN
      OUTPUT PRINTER
      SET LINES 56
    ENDIF
    SELECT MPI-ID PROJNAME=25 CAT=4 PRIORITY=4 DEVACT SCHEDDUE FROM PROJECT +
          SORT BY MPI-ID WHERE SYSTYPE = .CH1 AND STATUS = O

```

```
OUTPUT SCREEN
SET LINES 20
ENDIF
RETURN
```

```
*( ====== PROJLOAD.CMD ======
PROGRAM: PROJLOAD.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 30 DECEMBER 1985
PURPOSE: THIS PROGRAM ADDS A NEW PROJECT TO THE DATABASE
          USING THE VARIABLE VALUES INPUT INTO THE VARIABLE FORM
          PROJECT. ERROR CHECKING IS ENFORCED BY
          THE RULES EXISTING IN THE DATA DICTIONARY.

TABLES USED: PROJECT
VARIABLE FORMS USED: PROJECT
=====

SET NULL -0-
SET VAR $OKAY = FALSE      *( initialize load flag to false )
WRITE AT 1 1                " +
* ( load the new record into the system table)
LOAD PROJECT
  .VMID : VPID : VSYS : VCAT : VPROJ + VSTAT . VDEV . VPROJ +
  .VSWT : VREQ : VDUE : VSCHED . VCOMP . VTEST
END
SET VAR CHECK = MPI-ID IN PROJECT WHERE COUNT = LAST
IF CHECK = VMID THEN
  SET VAR $OKAY = TRUE
NEWPAGE
WRITE "NEW RECORD ADDED FOR PROJECT: "
SHOW VAR VMID AT 1 31 { show user the record added }
WRITE "PRESS ANY KEY TO CONTINUE . . ." AT 10 10
PAUSE
ELSE
  *( reset original priorities )
  IF VPRI EXISTS THEN
    ASSIGN PRIORITY TO VPRI +
    AND DEVACT = .VDEV
  ENDIF
  WRITE "PRESS ANY KEY TO CONTINUE . . ." AT 24 10
PAUSE
NEWPAGE
ENDIF
RETURN
```

Option 1 will add a new system, a new document in the life cycle documentation package or a new development activity.  
 Option 2 provides the capability to change the attributes of a system, a particular document or information about a development activity.  
 Option 3 allows deletion of a system, a document or a development activity.  
 Option 4 provides different retrievals from the database by system, document and development activity. See the User's Manual for more details.  
 Option 5 prints standard reports to the output device that is designated by the user.

```
*( =====
  PROGRAM: RBASE.DAT
  AUTHOR : R.C. CAVALLARO
  DATE WRITTEN: 9 FEBRUARY 1986
  PURPOSE: THIS PROGRAM AUTOMATICALLY SETS THE COLOR ATTRIBUTES AND
            STARTS THE MCAS MIS SYSTEM WHEN RBASE IS INVOKED. DISALLOWS
            USER ACCESS TO THE RBASE PROGRAMMING LANGUAGE.
  =====)
SET COLOR FORE GRAY
SET COLOR BACK BLUE
RUN MAIN.CMD
EXIT

*( =====
  PROGRAM: SOOPICK.CMD
  AUTHOR : R.C. CAVALLARO
  DATE WRITTEN: 17 NOVEMBER 1985
  PURPOSE: THE MAIN MENU OF THE SYSTEMS MANAGEMENT SUBSYSTEM. ROUTES
            USER TO PROPER MENU BASED ON INDICATED INPUT TO MENU.
  PROGRAMS CALLED: S01PICK.CMD, S02PICK.CMD, S03PICK.CMD,
  SCREENS DISPLAYED: S00HLP, SMENUOO
  =====)

*( initialize loop control variable
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
  NEWPAGE
  WRITE "SELECT ANY OPTION WITH NUMBER OR CURSOR ARROW AND ENTER"
  WRITE "F10 FOR HELP" AT 1 66
  WRITE
  CHOOSE "CHOICE FROM SMENUOO" . PLEASE *( systems management main menu )
  WRITE "INPUT ACCEPTED" .
```

```

IF CHOICE = -1 THEN          *( F10 was pressed for help )
  DISPLAY SOOHLP
  WRITE "PRESS ANY KEY TO CONTINUE . . ."
  PAUSE
  SET VAR CHOICE = 0
ENDIF

IF CHOICE = 1 THEN          *( add menu )
  RUN SO1PICK.CMD
ENDIF

IF CHOICE = 2 THEN          *( edit menu )
  RUN SO2PICK.CMD
ENDIF

IF CHOICE = 3 THEN          *( delete menu )
  RUN SO3PICK.CMD
ENDIF

IF CHOICE = 4 THEN          *( query menu )
  RUN SO4PICK.CMD
ENDIF

IF CHOICE = 5 THEN          *( management reports menu )
  RUN SO5PICK.CMD
ENDIF

IF CHOICE = 6 THEN          *( return to MCASMS main menu )
  BREAK
ENDIF

SET VAR CHOICE = 0
ENDWHILE
SET VAR CHOICE = 0
RETURN

```

Option 1 will provide a form for input of appropriate fields in the system record. After completing the form, the user should press PGDN to load the data or Escape to quit without loading. ENTER tabs to next input area. The user is only prompted for pertinent input based on the category that is input in the first part of the record. Valid categories are MCD, MCD, DNO and DNE. Option 2 will provide a similar function for adding a new document. Again, press PGDN to load the data and Escape to quit without loading. The ENTER key again serves as the tab to the next input field. Option 3 provides a form for loading a new development activity. The PGDN, Enter and Escape keys provide the same functions as above.

```

*( =====
PROGRAM: S01PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 17 NOVEMBER 1985
PURPOSE: SYSTEMS MANAGEMENT ADD MENU. DISPLAYS PROPER FORM FOR
          ADDING A SYSTEM, DOCUMENT OR DEVELOPMENT ACTIVITY BASED
          ON USER'S INPUT TO MENU. ERROR CHECKING IS ENFORCED BY
          THE RULES EXISTING IN THE DATA DICTIONARY.

PROGRAM CALLED: MAIN.CMD, S11PICK.CMD, S21PICK.CMD, S31PICK.CMD
SCREENS DISPLAYED: S01HLP, SMENU01
=====)

*(initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
  NEWPAGE
  WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
  WRITE "F10 FOR HELP" AT 1 66
  WRITE "CHOOSE " CHOICE FROM SMENU01 *(from systems mgmt add menu)
  WRITE "INPUT ACCEPTED . . PLEASE WAIT."
  IF CHOICE = -1 THEN *( F10 pressed for help)
    DISPLAY S01HLP
    WRITE "PRESS ANY KEY TO CONTINUE . . . "
    PAUSE
  ENDIF

  IF CHOICE = 1 THEN *( add a new system)
    RUN S11PICK.CMD
  ENDIF

  IF CHOICE = 2 THEN *( add a new document)
    RUN S21PICK.CMD
  ENDIF

  IF CHOICE = 3 THEN *( add a new development activity)
    RUN S31PICK.CMD
  ENDIF

  IF CHOICE = 4 THEN *( return to systems management main menu )
    BREAK
  ENDIF

  IF CHOICE = 5 THEN *( quit to MCASMIS main menu )
    QUIT TO MAIN.CMD
  ENDIF

```

```
SET VAR CHOICE = 0      *( redisplay add menu )
ENDWHILE
SET VAR CHOICE = 0      *( reset loop control variable)
RETURN
```

Option 1: after changing the requested system, press PGDN to load,  
ENTER to tab to next input area, PGUP to find next matching record,  
or ESC to quit without changing.  
Option 2: on the presented form press ENTER to tab to next input field, PGDN  
to load the changes, ESC to quit without loading, or PGUP to find the  
next matching record without loading the current changes.  
Option 3: on the presented form press ENTER to tab to next input area, PGDN  
to load the changes or ESC to quit without loading.

```
*=====
PROGRAM: S02PICK.CMD
AUTHOR: R. CAVALLARO
DATE WRITTEN: 17 NOVEMBER 1985
PURPOSE: SYSTEMS MANAGEMENT EDIT MENU. DISPLAYS PROPER FORM FOR
EDITING A SYSTEM DOCUMENT OR DEVELOPMENT ACTIVITY BASED
ON USER'S INPUT TO MENU. ERROR CHECKING IS ENFORCED BY
THE RULES EXISTING IN THE DATA DICTIONARY.

PROGRAM CALLED: S12PICK.CMD, S22PICK.CMD, S32PICK.CMD, MAIN.CMD
SCREENS DISPLAYED: SMENU02, S02HLP
=====
```

```
*(initialize loop control variable)
SET VAR CHOICE = 0
WHILE CHOICE = 0 THEN
NEWPAGE
WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
WRITE "F10 FOR HELP" AT 1 66
WRITE "1" TO CONTINUE . . .
CHOICE FROM SMENU02
WRITE "INPUT ACCEPTED . PLEASE WAIT."
IF CHOICE = -1 THEN      *( F10 pressed for help)
DISPLAY S02HLP
WRITE "PRESS ANY KEY TO CONTINUE . . .
PAUSE
ENDIF
IF CHOICE = 1 THEN      *( edit the system )
ENDIF
```

```

IF CHOICE = 2 THEN          *( edit a document )
  RUN S22PICK.CMD
ENDIF

IF CHOICE = 3 THEN          *( edit a development activity )
  RUN S32PICK.CMD
ENDIF

IF CHOICE = 4 THEN          *( return to systems management main menu )
  BREAK
ENDIF

IF CHOICE = 5 THEN          *( quit to MCASMSI main menu )
  QUIT TO MAIN.CMD
ENDIF

SET VAR CHOICE = 0          *( redisplay edit menu )

ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0          *( reset loop control variable )
RETURN

```

Option 1 deletes a system from the database.  
 Option 2 deletes a document from the database.  
 Option 3 deletes a development activity from the database.

In all options, the user is first presented the information in the appropriate form to verify that the record should be deleted. Press PGDN to delete the record shown, PGUP to find the next match or ESC to quit and return to the last menu. The process of deleting a record requires confirmation and the user is given one additional chance to quit without changing in case a wrong key was pressed.

```

*( ====== PROGRAM: S03PICK.CMD ======
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 17 NOVEMBER 1985
  PURPOSE: SYSTEMS MANAGEMENT DELETE MENU. DELETES THE REQUESTED RECORDS
           FROM THE DATA BASE BY SYSTEM DOCUMENT OR DEVELOPMENT ACTIVITY
           BASED UPON USER'S INPUT TO MENU.

  PROGRAM CALLED: S13PICK.CMD S23PICK.CMD, S33PICK.CMD, MAIN.CMD
  SCREENS DISPLAYED: S03HLP $MENU03
  TABLES USED: SYSTEM, DOCLIST, VENDOR
  VARIABLE FORMS USED: SYS, Doc, VEND
=====)

```

```

SET VAR CHOICE = 0          *(* initialize loop control variable )
WHILE CHOICE = 0 THEN
    NEWPAGE
    WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
    WRITE "F10 FOR HELP AT 1 66
    WRITE "H
    CHOOSE CHOICE FROM SMENU03
    WRITE "INPUT ACCEPTED . . PLEASE WAIT."
    WRITE "H

    IF CHOICE = -1 THEN      *(* F10 pressed for help )
        DISPLAY S03HLP
        WRITE "PRESS ANY KEY TO CONTINUE . . .
        PAUSE
    ENDIF

    IF CHOICE = 1 THEN      *(* delete a system )
        RUN S13PICK.CMD
    ENDIF

    IF CHOICE = 2 THEN      *(* delete a document )
        RUN S23PICK.CMD
    ENDIF

    IF CHOICE = 3 THEN      *(* delete a development activity )
        RUN S33PICK.CMD
    ENDIF

    IF CHOICE = 4 THEN      *(* return to systems management main menu )
        BREAK
    ENDIF

    IF CHOICE = 5 THEN      *(* quit to MCASMSI main menu )
        QUIT TO MAIN.CMD
    ENDIF

    SET VAR CHOICE = 0          *(* redisplay delete menu )
ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0
RETURN

```

The user is provided with selected information based on the option selected and the answers to the subsequent prompts. Options 4, and 6 will provide output on any record that contains the response to the prompt. For example, if you are uncertain about the correct abbreviation for "ideamatics, Inc.", try "idea" as input to the prompt for the development activity. Upper and lower case distinctions are ignored. For example, selections on ABC, abc, or ABC will produce the same output.

Output is routed to the screen unless the user has selected option 1 immediately preceding the query. EXCEPT that option 7 may not be routed directly to the printer.

```

*=====
* PROGRAM: SO4PICK.CMD
* AUTHOR : R. C. CAVALLARO
* DATE WRITTEN: 17 NOVEMBER 1985
* PURPOSE: OUTPUTS SELECTED INFORMATION DESIRED BY USER AFTER DISPLAYING
*          SELECTION CRITERIA

PROGRAM CALLED : SYSDRAW.CMD, MAIN.CMD, S24PICK.CMD, S34PICK.CMD,
SCREENS DISPLAYED: S04HLP, SMENU04
TABLES USED: SYSTEM

=====
* SET NULL " " *( display null values as blanks )
* SET VAR IO = N *( initialize loop control variable )
* SET VAR CHOICE = 0 *{ set error variable for trapping
* SET ERROR VAR ERV = 0 THEN
WHILE CHOICE = 0 THEN
NEWPAGE
WRITE "SELECT ANY OPTION USING NUMBER OR CURSOR ARROW AND ENTER"
WRITE "F10 FOR HELP AT 1 66
WRITE "W
CHOOSE "CHOICE FROM SMENU04
WRITE "INPUT ACCEPTED . . PLEASE WAIT."
WRITE

IF CHOICE = -1 THEN *( F10 pressed for help )
DISPLAY S04HLP
ENDIF

IF CHOICE = 1 THEN *( change output device )
RUN TOPICK.CMD
ENDIF

IF CHOICE = 2 THEN *( customized tally command )
RUN S24PICK.CMD
ENDIF

IF CHOICE = 3 THEN *( select documents by system )
RUN S34PICK.CMD
ENDIF

IF CHOICE = 4 THEN *( select systems by document type )
RUN S44PICK.CMD
ENDIF

```

```

ENDIF
IF CHOICE = 5 THEN      *( on-line data dictionary )
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  LIST COLUMNS
  LIST ALL
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF CHOICE = 6 THEN      *( docs and systems by development activity )
  IF RUN S64PICK.CMD
  ENDIF

IF CHOICE = 7 THEN      *( select all info about a system )
  FILLIN CH1 USING "ENTER THE SYSTEM ABBREVIATION"
  SET POINTER #1 PT1 FOR SYSTEM WHERE SYSSNAME = .CH1
  IF PT1 NE 0 THEN
    WRITE "THERE IS NO SYSTEM ON FILE BY THAT NAME"
  ELSE
    NEWPAGE
    SET NULL "-O-"
    RUN SYSDRAW.CMD
    SET NULL
  ENDIF
ENDIF

IF CHOICE = 8 THEN      *( return to systems management main menu )
  BREAK
ENDIF

IF CHOICE = 9 THEN      *( return to MCASMIS main menu )
  QUIT TO MAIN.CMD
ENDIF

IF CHOICE NE 1 THEN
  WRITE "PRESS ANY KEY TO CONTINUE . . . "
  PAUSE
ENDIF
SET VAR CHOICE = 0
ENDWHILE
CLEAR ALL VAR
SET VAR CHOICE = 0
SET NULL "-O-"
RETURN
*{ reset loop control variable
*{ reset default null indicator

```

CHG OUTPUT  
 DOCS DUE  
 TASK DOCS  
 PEND DOCS  
 PEND PROJS  
 NEED DATA  
 LIST SYSTS  
 LIST DOCS  
 LIST VENDS  
 ANALYSTS  
 SYSTEMS  
 OPER VIEW  
 DEVEL VIEW  
 HELP  
 RETURN  
 EXIT

Select output device for following reports.  
 Lists MENS and SDP documents coming due or overdue.  
 Lists MENS and SDP with due dates that are not yet assigned.  
 Lists all documents that are pending (not completed).  
 Lists all open projects for a specified system.  
 Flags selected data elements that are missing from the operational systems in the database.  
 Lists all systems sorted by category and system name.  
 Lists all documents in the database.  
 Lists all development activity names and phone numbers.  
 Lists system names and analysts sorted by analyst name.  
 Lists system names and analysts sorted by system name.  
 Lists overview data on operational systems.  
 Lists overview data on development/enhancement systems.  
 Displays this help file for on-line assistance.  
 Returns control to previous menu.  
 Returns control to MCASMS main menu.

```

*( =====
PROGRAM: S05PICK.CMD
AUTHOR : R. CAVALLARO
DATE WRITTEN: 29 NOVEMBER 1985
PURPOSE: THIS PROGRAM PROVIDES VARIOUS INFORMATION PERTAINING TO THE
MANAGEMENT OF SYSTEMS.

PROGRAM CALLED: IOPICK.CMD, MAIN.CMD, S15PICK.CMD, S25PICK.CMD,
S35PICK.CMD, PROJLIST.CMD
SCREENS DISPLAYED: S05HLP.CMD, SMENU05
TABLES USED: SYSTEM, DOCLIST, VENDOR
=====)

SET NULL " "
*( set null to blank for display )
SET VAR IO = N
SET VAR TGT DATE
SET VAR DAYS INTEGER
SET VAR PICK = GO
*( loop control variable )

WHILE PICK = GO THEN
NEWPAGE
WRITE "SELECT ANY OPTION USING CURSOR KEYS AND ENTER"
WRITE "F10 FOR HELP AT 166"
WRITE "CHOOSE PICK FROM SMENU05"
WRITE "INPUT ACCEPTED . . PLEASE WAIT."
WRITE

IF PICK = HELP THEN
  *( F10 pressed for help )

```

```

NEWPAGE
DISPLAY S05HLP
ENDIF

IF PICK = "CHG OUTPUT" THEN
RUN IOPICK.CMD
ENDIF

IF PICK = "DOCS DUE" THEN
RUN S15PICK.CMD
ENDIF

IF PICK = "TASK DOCS" THEN
RUN S25PICK.CMD
ENDIF

IF PICK = "PEND DOCS" THEN
SET POINTER #1 FOR DOCLIST WHERE DSTAT = P
IF IO = Y THEN
OUTPUT SCREEN WITH PRINTER
SET LINES 56
ENDIF
IF PT1 NE O THEN
WRITE "THERE ARE NO PENDING DOCUMENTS AT THIS TIME"
ELSE
NEWPAGE
SELECT SYSNAME DOCABR DEVACT FROM DOCLIST SORTED BY SYSNAME +
WHERE DSTAT = P
ENDIF
OUTPUT SCREEN
SET LINES 20
ENDIF

IF PICK = "NEED DATA" THEN
RUN S35PICK.CMD
ENDIF

IF PICK = "LIST SYSTS" THEN
NEWPAGE
IF IO = Y THEN
OUTPUT SCREEN WITH PRINTER
SET LINES 56
ENDIF
SELECT SYSNAME CATEGORY LNGTITLE FROM SYSTEM SORT BY CATEGORY=D SYSNAME
OUTPUT SCREEN
SET LINES 20
ENDIF

IF PICK = "LIST DOCS" THEN
NEWPAGE

```

```

IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
SELECT SYSNAME DOCABR=6 DEVACT=12 DOCNAME=30 DOCDATE FROM DOCLIST +
  SORTED BY SYSNAME
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF PICK = "LIST VENDS" THEN
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT DEVACT DEVLONG=50 PHONE FROM VENDOR SORTED BY DEVACT
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF PICK = "ANALYSTS" THEN
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT ANALYST SYSNAME CATEGORY LNGTITLE FROM SYSTEM SORTED BY +
    ANALYST SYSNAME
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF PICK = "SYSTEMS" THEN
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT SYSNAME ANALYST CATEGORY LNGTITLE FROM SYSTEM SORTED BY +
    SYSNAME
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF PICK = "OPER VIEW" THEN
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF

```

```

ENDIF
SELECT SYSDNAME ENV CLASS IMPDATE PRIUSER COMPLANG PGMS=4 LOC=4 +
FROM SYSTEM SORTED BY SYSDNAME WHERE CATEGORY = MCO
OUTPUT SCREEN
SET LINES 20
ENDIF

IF PICK = "DEVEL VIEW" THEN
  NEWPAGE
  IF IO = Y THEN
    OUTPUT SCREEN WITH PRINTER
    SET LINES 56
  ENDIF
  SELECT SYSDNAME MENS SDPIL SDPILL EXPLIIE=3 PRIUSER FROM +
  SYSTEM SORT BY CATEGORY SYSDNAME WHERE CATEGORY = MCD OR +
  CATEGORY = MCE
  OUTPUT SCREEN
  SET LINES 20
ENDIF

IF PICK = "PEND PROJS" THEN
  RUN PROJLIST.CMD
ENDIF

IF PICK = "RETURN" THEN      *(* return to systems management main menu )
BREAK
ENDIF

IF PICK = "EXIT" THEN
  QUIT TO MAIN.CMD
ENDIF

IF PICK NE "CHG OUTPUT" THEN
  WRITE "PRESS ANY KEY TO CONTINUE . . . "
PAUSE
ENDIF
SET VAR PICK = GO
ENDWHILE
SET NULL -O-
CLEAR ALL VAR
SET VAR CHOICE = 0
RETURN

*( reset loop control variable )
*{ reset default null value )
*{ clear all variables )
*{ reset loop control variable )

*( =====
PROGRAM: S11PICK.CMD
AUTHOR: R. C. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: ADD A NEW SYSTEM.

```

PROGRAMS CALLED: SYSLOAD. CMD  
TABLES USED: SYSTEM  
VARIABLE FORMS USED: SYS

```
=====
SET VAR MORE TO "Y"          *( initialize loop control variable)
WHILE MORE = Y THEN
  NEWPAGE
  DRAW SYS           *( variable form SYS with null values
  WRITE "ENTER TO TAB TO NEXT FIELD PGDN TO LOAD ESC T) QUIT" +
  AT 1 1
  *( user inputs values in form and continues after pressing
    PGDN or ESCape)
  ENTER VAR VSYSVLONG VCAT* VANAL RETURN PGDN ESC
  IF #RETURN = ESC THEN * ( if user pressed ESC then quit )
    BREAK
  ENDIF
  *(if operational system, prompt for and enter the following)
  IF VCAT = MCO OR VCAT = DNO THEN
    ENTER VAR VPRIVENV VCLASS VLANG VPRNO VLOC VIMP RETURN PGDN ESC
  *{if not operational, prompt for dates, exp. life and user }
  ELSE
    ENTER VAR VMENS VII VIII VAMENS VAI VAI VAI VAI +
    VEXP VPRI RETURN PGDN ESC
  ENDIF
  IF #RETURN = ESC THEN      *( quit if ESC is pressed )
    BREAK
  ENDIF
  *(load the new record into the system table)
  RUN SYSLOAD.CMD
  *(prompt for another system addition
  FILIN MORE USING ADD ANOTHER SYSTEM { Y/N): " AT 11 25
  ENDWHILE
  SET VAR CHOICE = 0
  RETURN
```

```
=====
*( PROGRAM: S12PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 7 JANUARY 1986
  PURPOSE: EDIT A SYSTEM. AUTOMATIC RULE CHECKING IS ENFORCED
            SO THAT A VALUE MAY NOT BE CHANGED TO AN ILLEGAL ONE ( e.g.,
            A SYSTEM NAME MAY NOT BE CHANGED TO A NULL VALUE ).
```

PROGRAMS CALLED: SYSLOAD. CMD, SYSDRAW. CMD  
TABLES USED: SYSTEM

```

=====
)-----+
WRITE "ENTER THE SYSTEM ABBREVIATION OR ENTER L FOR A LIST OF ALL"
FILLIN CH1 USING CURRENT SYSTEM ABBREVIATIONS:
IF CH1 = L THEN
  SELECT "SYSNAME LNGTITLE=60 FROM SYSTEM SORT BY SYSNAME
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
  RETURN
ENDIF
SET POINTER #1 PT1 FOR SYSTEM WHERE SYSNAME = CH1
IF PT1 NE O THEN
  * ( matching record not found )
  WRITE "THIS SYSTEM IS NOT IN THE FILE . . YOU MUST ADD IT"
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ELSE
  NEWPAGE
  RUN SYSDRAW CMD *( variable form with values from record)
  WRITE "ENTER TO TAB
  AT 1 RETURN PGDN TO LOAD
  EDIT VAR RETURN PGDN ESC
  IF #RETURN = PGDN THEN
    * ( save the change if PGDN )
    RUN SYSLOAD CMD *( load the new row )
    IF OKAY = T THEN
      * ( if load was successful )
      DELETE ROWS FROM #1 *( delete the old row )
    ENDIF
  ENDIF
  ENDIF
  RETURN
* (
=====
)-----+
PROGRAM: S13PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: DELETE A SYSTEM. PROHIBITS USER FROM DELETING A SYSTEM
FOR WHICH THERE ARE OPEN PROJECTS ON FILE.

PROGRAMS CALLED: SYSDRAW.CMD
TABLES USED: PROJECT, SYSTEM
=====

)-----+
FILLIN CH1 USING "WHICH SYSTEM ABBREVIATION TO DELETE:
SET POINTER #1 PT1 FOR SYSTEM WHERE SYSNAME = CH1
IF PT1 NE O THEN
  * ( no such system name
  WRITE "THIS SYSTEM IS NOT IN THE FILE - IT CANNOT BE DELETED"
=====

)

```

```

        WRITE " "
        WRITE "PRESS ANY KEY TO CONTINUE . . ."
        PAUSE
        ELSE
          SET POINTER #2 PT2 * FOR PROJECT WHERE SYSNAME = CH1 AND STATUS = 0
          IF PT2 NE 0 THEN *( no open projects for this system )
            RUN SYSDRAW. CMD *( draw sys form with values of matching record )
            WRITE "PGDN TO DELETE AT 1 1
            EDIT VAR VSYS RETURN PGDN ESC
            IF #RETURN = PGDN THEN *( delete the record )
              NEWPAGE
              WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
              FILLIN ANS USING RECORD *{ Y/N}
              IF ANS = Y THEN *( after one last chance )
                DELETE ROWS FROM #1
              ENDIF
            ENDIF
          ELSE
            WRITE " "
            WRITE "YOU MAY NOT DELETE THIS SYSTEM UNTIL THE OPEN PROJECTS FOR"
            WRITE "IT HAVE BEEN EITHER CLOSED OR DELETED."
            WRITE "PRESS ANY KEY TO CONTINUE . . ."
            PAUSE
          ENDIF
        ENDIF
        RETURN
      
```

```

*( =====
  PROGRAM: S15PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 29 NOVEMBER 1985
  PURPOSE: PROMPTS USER FOR A TARGET NUMBER OF DAYS AND THEN DISPLAYS
           ALL MENS AND SDPI'S THAT ARE DUE WITHIN THAT PERIOD.
=====)

```

TABLE USED: SYSTEM

```

RUN ELXDATE. CMD
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
WRITE "THE FOLLOWING MENS DOCUMENTS WILL BE DUE OR ARE OVERDUE"
WRITE

```

```

SELECT SYSCNAME MENS FROM SYSTEM SORTED BY MENS WHERE MENS LE . TGT +
AND AMENS FAILS
IF IO NE Y THEN
  WRITE " "
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ENDIF "
WRITE "
WRITE "THE FOLLOWING SYSTEMS DECISION PAPERS WILL BE DUE OR ARE OVERDUE"
WRITE "
WRITE "SDPI"
WRITE "
SELECT SYSCNAME SDPI FROM SYSTEM SORTED BY SDPI WHERE SDPI LE . TGT +
AND ASDPI FAILS
IF IO NE Y THEN
  WRITE " "
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ENDIF "
WRITE "
WRITE "SDPII"
WRITE "
SELECT SYSCNAME SDPII FROM SYSTEM SORTED BY SDPII WHERE SDPII LE . TGT +
AND ASDPII FAILS
IF IO NE Y THEN
  WRITE " "
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ENDIF "
WRITE "
WRITE "SDPIII"
WRITE "
SELECT SYSCNAME SDPIII FROM SYSTEM SORT BY SDPIII WHERE SDPIII LE . TGT +
AND ASDPIII FAILS
OUTPUT SCREEN
SET LINES 20
RETURN

```

\*(-----  
PROGRAM: S21PICK.CMD  
AUTHOR : R.C.CAVALLARO  
DATE WRITTEN: 7 JANUARY 1986  
PURPOSE: ADD A NEW DOCUMENT TO THE FILE AUTOMATIC ERROR CHECKING  
IS ENFORCED AS EXPLAINED IN THE USERS MANUAL.

TABLES USED: DOCLIST  
VARIABLE FORMS USED: DOC

```

SET VAR MORE TO "Y"          * ( initialize loop control )
WHILE MORE = Y THEN
  NEWPAGE
  DRAW DOC
  WRITE "ENTER TO TAB TO NEXT FIELD" * ( variable form with null values ) TO QUIT" +
  AT 1 1
  * ( prompt for and enter the following variables )
  ENTER VAR VSYS VDOC VTIT VACT VDATE VSTAT RETURN PGDN ESC
  IF #RETURN = ESC THEN
    IF #RETURN = ESC THEN * ( quit if user pressed ESC )
      BREAK
    ENDIF
  * ( load the new record into the doclist table )
  SET POINTER #1 PT1 FOR DOCLIST WHERE COUNT = LAST
  WRITE " +"
  SET POINTER #1 PT1 FOR DOCLIST WHERE COUNT = LAST
  WRITE " +"
  SET ERROR VAR ERV
  LOAD DOCLIST
  VSYS . VDOC . VTIT . VACT . VDATE . VSTAT
  END
  IF ERV = 0 THEN          * ( the record was added successfully )
    NEWPAGE
    WRITE "NEW RECORD ADDED FOR ACTIVITY: "
    SHOW VAR VACT AT 1 33 * ( show user the record added )
  ELSE
    WRITE "PRESS ANY KEY TO CONTINUE . . ." AT 23 10
    PAUSE
    NEWPAGE
  ENDIF
  CLEAR ALL VAR
  * ( prompt for another document addition )
  FILLIN MORE USING ADD ANOTHER DOCUMENT { Y/N ): " AT 11 25
ENDWHILE
SET VAR CHOICE = 0
RETURN

```

CO

---

```

*( ===== PROGRAM: S22PICK.CMD =====
  AUTHOR : R. CAVALLARO
  DATE WRITTEN: 7 JANUARY 1986
  PURPOSE: EDIT A DOCUMENT. AUTOMATIC RULE CHECKING IS ENFORCED.

  TABLES USED: DOCLIST
  VARIABLE FORMS USED: DOC
===== )

```

FILLIN CH1 USING "ENTER THE SYSTEM ABBREVIATION: "
FILLIN CH2 USING "ENTER THE DOCUMENT ABBREVIATION: "

```

*( the following command uses CONTAINS rather than " = " since
some documents are classified as more than one document type )
SET POINTER #1 PT1 FOR DOCLIST WHERE DOCABR CONTAINS .CH2 AND +_
SYSNAME = CH1
IF PT1 NE 0 THEN *( no matching record found )
  WRITE "THIS DOCUMENT IS NOT ON FILE"
  WRITE "PLEASE CHECK AND REINPUT"
  WRITE "PRESS ANY KEY TO CONTINUE"
PAUSE
ELSE
  WHILE PT1 = 0 THEN *( while matching record is found )
    SET VAR VSYS TO SYSNAME IN #1
    SET VAR VDOC TO DOCABR IN #1
    SET VAR VTIT TO DOCNAME IN #1
    SET VAR VACTE TO DEVACT IN #1
    SET VAR VDATE TO DOCDATE IN #1
    SET VAR VSTAT TO DSTAT IN #1
    NEWPAGE " " *( display null values as a blank )
    DRAW DOC WITH ALL
    SET NULL -O- *( return null designator ) ESC TO QUIT" +
    WRITE "ENTER TO TAB TO NEXT FIELD PGDN TO LOAD" + AT 1 1
    WRITE "PGUP FOR NEXT MATCH" AT 1 60
    *{ edit all fields }
    EDIT VAR RETURN ESC PGDN PGUP
    IF #RETURN = ESC THEN *( quit if ESC was pressed )
      BREAK
    ENDIF
    IF #RETURN = PGDN THEN *( save changes if PGDN pressed )
      CHANGE SYSNAME TO VSYS IN #1
      CHANGE DOCABR TO VDOC IN #1
      CHANGE DOCNAME TO VTIT IN #1
      CHANGE DEVACT TO VACTE IN #1
      CHANGE DOCDATE TO VDATE IN #1
      CHANGE DSTAT TO VSTAT IN #1
    ENDIF
    NEXT #1 PT1 *( find next matching record )
  ENDWHILE
ENDIF
RETURN
* ( ====== S23PICK.CMD
  AUTHOR : R.C.CAVALLARO
  DATE WRITTEN: 7 JANUARY 1986
  PURPOSE: DELETE A DOCUMENT.

```

TABLES USED: DOCLIST  
VARIABLE FORMS USED: DOC

```
=====
FILLIN CH2 USING "ENTER DOCUMENT ABBREVIATION: "
FILLIN CH1 USING "ENTER SYSTEM ABBREVIATION: "
SET POINTER #1 PT1 FOR DOCLIST WHERE DOCABR = .CH2 AND SYSNAME = .CH1
IF PT1 NE 0 THEN
  * ( no such document )
  WRITE "DOCUMENT NOT ON FILE - CANNOT BE DELETED"
  WRITE "D"
  WRITE "PRESS ANY KEY TO CONTINUE"
  PAUSE
ELSE
  WHILE PT1 = 0 THEN
    SET VAR VSYS TO SYSNAME IN #1
    SET VAR VDOC TO DOCABR IN #1
    SET VAR VTIT TO DOCNAME IN #1
    SET VAR VACT TO DEVACT IN #1
    SET VAR VDATE TO DOCDATE IN #1
    SET VAR VSTAT TO DSTAT IN #1
    NEWPAGE
    SET NULL " " * ( reset null to blank for display )
    DRAW DOC WITH ALL
    SET NULL -0-* ( reset default null value indicator
    WRITE "PGDN TO DELETE PGUP FOR NEXT MATCH" AT 1 1
    WRITE "ESC TO QUIT" AT 1 60
    EDIT VAR VSYS RETURN PGDN ESC PGUP
    IF #RETURN = PGDN THEN
      NEWPAGE
      WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
      FILLIN ANS USING "RECORD (Y/N)"
      IF ANS = Y THEN
        DELETE ROWS FROM #1 * ( after one last chance )
      ENDIF
      IF #RETURN = ESC THEN
        * ( quit if ESC pressed )
        BREAK
      ENDIF
      NEXT #1 PT1
    ENDWHILE
  ENDIF
  RETURN
=====
```

\* ( =====  
PROGRAM: S24PICK.CMD  
AUTHOR: R. C. CAVALLARO  
DATE WRITTEN: 17 NOVEMBER 1985  
PURPOSE: ALLOWS THE USER TO GET CUSTOMIZED TALLY INFORMATION

CONCERNING NUMBER OF OCCURRENCES IN THE DATABASE.

```
=====)
SET ERROR VARIABLE ERV
NEWPAGE
WRITE " WHICH OF THE FOLLOWING TABLES DO YOU WISH TO ACCESS"
WRITE " "
LIST TABLES
WRITE " "
FILLIN CH1 USING "ENTER THE TABLE NAME: "
NEWPAGE
LIST .CH1
WRITE " "
FILLIN CH2 USING "WHICH COLUMN SHOULD BE TALLIED: "
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
TALLY .CH2 IN CH1
IF ERV NE 0 THEN
  OUTPUT SCREEN
  WRITE "INVALID TABLE OR COLUMN NAME WAS ENTERED; PLEASE CHECK YOUR"
  WRITE "INPUT AND TRY AGAIN."
ENDIF
OUTPUT SCREEN
SET LINES 20
SET VAR CHOICE = 0
RETURN
*( redisplay query menu )
```

```
*( =====)
PROGRAM: S25PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 29 NOVEMBER 1985
PURPOSE: DISPLAYS ALL MENS AND SDPI'S THAT HAVE DUE DATES IN THE FILE
          BUT HAVE NOT YET BEEN TASKED TO ANYONE TO COMPLETE.

TABLES USED: SYSTEM, DOCLIST
=====)
```

```
SET POINTER #1 PT1 FOR SYSTEM SORTED BY MENS WHERE MENS EXISTS +
AND AMENS FAILS
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
WRITE "MENS"
```

```

      WRITE " "
      IF PT1 NE 0 THEN
        WRITE "ALL MENS DOCUMENTS ON FILE HAVE BEEN COMPLETED"
      ELSE
        WHILE PT1 = 0 THEN
          SET VAR TSYS = SYSNAME IN #1
          SET VAR TDOC = MENS IN #1
          SET POINTER #2 PT2 FOR DOCLIST WHERE SYENAME = .TSYS AND +
            DOCABR = .TDOC AND DSTAT = P
          IF PT2 NE 0 THEN
            SET VAR NAME = .TSYS & .TDOC
            SHOW VAR NAME
          ENDIF
        NEXT #1 PT1
      ENDIF
      IF IO NE Y THEN
        WRITE "PRESS ANY KEY TO CONTINUE . . "
        PAUSE
      ENDIF
      WRITE " "
      SET POINTER #1 PT1 FOR SYSTEM SORTED BY SDPI WHERE SDPI EXISTS +
        AND ASDPI FAILS
      WRITE "SDPI"
      WRITE ""
      IF PT1 NE 0 THEN
        WRITE "ALL SDPI DOCUMENTS THAT HAVE DUE DATES HAVE BEEN COMPLETED"
      ELSE
        WHILE PT1 = 0 THEN
          SET VAR TSYS = SYSNAME IN #1
          SET VAR TDOC = SDPI IN #1
          SET POINTER #2 PT2 FOR DOCLIST WHERE DSTAT = P AND +
            DOCABR = .TDOC AND SYENAME = .TSYS
          IF PT2 NE 0 THEN
            SET VAR NAME = .TSYS & .TDOC
            SHOW VAR NAME
          ENDIF
        NEXT #1 PT1
      ENDIF
      IF IO NE Y THEN
        WRITE "PRESS ANY KEY TO CONTINUE . . "
        PAUSE
      ENDIF
      WRITE " "
      SET POINTER #1 PT1 FOR SYSTEM SORTED BY SDPII WHERE SDPII EXISTS +
        AND ASDPII FAILS
      WRITE "SDPII"

```

```

        WRITE " "
        IF PT1 NE 0 THEN WRITE "ALL SDP III DOCUMENTS THAT HAVE DUE DATES HAVE BEEN COMPLETED"
        ELSE WHILE PT1 = 0 THEN
                SET VAR TSYS = SYSSNAME IN #1
                SET VAR TDOC = SDP III IN #1
                SET POINTER #2 PT2 FOR DOCLIST WHERE DSTAT = P AND +
                DOCABR = .TDOC AND SYSSNAME = .TSYS
                IF PT2 NE 0 THEN
                        SET VAR NAME = .TSYS & .TDOC
                        SHOW VAR NAME
                ENDIF
                NEXT #1 PT1
        ENDWHILE
        ENDIF
        IF IO NE Y THEN
                WRITE "PRESS ANY KEY TO CONTINUE . . "
                PAUSE
        ENDIF
        WRITE " "
        SET POINTER #1 PT1 FOR SYSTEM SORTED BY SDP III WHERE SDP III EXISTS +
        AND ASDP III FAILS
        WRITE "SDP III"
        WRITE " "
        IF PT1 NE 0 THEN WRITE "ALL SDP III DOCUMENTS THAT HAVE DUE DATES HAVE BEEN COMPLETED"
        ELSE WHILE PT1 = 0 THEN
                SET VAR TSYS = SYSSNAME IN #1
                SET VAR TDOC = SDP III IN #1
                SET POINTER #2 PT2 FOR DOCLIST WHERE DSTAT = P AND +
                DOCABR = .TDOC AND SYSSNAME = .TSYS
                IF PT2 NE 0 THEN
                        SET VAR NAME = .TSYS & .TDOC
                        SHOW VAR NAME
                ENDIF
                NEXT #1 PT1
        ENDWHILE
        ENDIF
        OUTPUT SCREEN
        SET LINES 20
        RETURN

```

\* ( ====== PROGRAM: S31PICK.CMD  
 AUTHOR: R.C. CAVALLARO  
 DATE WRITTEN: 7 JANUARY 1986

PURPOSE: ADD A NEW DEVELOPMENT ACTIVITY.

TABLES USED: VENDOR  
VARIABLE FORMS USED: VEND

```
=====
SET VAR MORE TO "Y"          * ( initialize loop control )
WHILE MORE = Y THEN
NEWPAGE
DRAW VEND
WRITE "ENTER TO TAB TO NEXT FIELD" +  
AT 1_1 *( variable form with null values ) TO QUIT" +
*( prompt for and enter these values )
ENTER VAR VACT VFULL VPOC VPH VADR VCITY VST VZIP RETURN PGDN ESC
IF #RETURN = ESC THEN      *( quit if user pressed ESC )
BREAK
ENDIF
WRITE "
WRITE AT 1_1
SET ERROR VAR ERV
*( load the new record into the vendor table )
LOAD VENDOR
VACT .VFULL .VPOC .VPH .VADR .VCITY .VST .VZIP
END
IF ERV = 0 THEN
NEWPAGE
WRITE NEW RECORD ADDED FOR ACTIVITY:
SHOW VAR VACT AT 1_33 *( show user the record added )
ELSE
WRITE "PRESS ANY KEY TO CONTINUE . . ." AT 23 10
PAUSE
NEWPAGE
ENDIF
CLEAR ALL VAR
*( prompt for another activity addition { Y/N): " AT 11 25
FILLIN MORE USING "ADD ANOTHER ACTIVITY"
ENDWHILE
SET VAR CHOICE = 0
RETURN
=====
```

\*( ====== PROGRAM: S32PICK.CMD  
AUTHOR: R. C. CAVALLARO  
DATE WRITTEN: 7 JANUARY 1986  
PURPOSE: EDIT A DEVELOPMENT ACTIVITY.

TABLES USED: VENDOR  
VARIABLE FORMS USED: VEND

```

FILLIN CH1 USING "ENTER THE DEVELOPMENT ACTIVITY ABBREVIATION:
SET PTR #1 PT1 FOR VENDOR WHERE DEVACT = .CH1
IF PTR NE 0 THEN *( no matching record found )
    WRITE "THIS DEVELOPMENT ACTIVITY IS NOT ON FILE"
    WRITE "PLEASE CHECK AND REINPUT"
    WRITE "PRESS ANY KEY TO CONTINUE"
PAUSE
ELSE
    SET VAR VACT TO DEVACT IN #1
    SET VAR VFULL TO DEVLONG IN #1
    SET VAR VPOC TO CONTACT IN #1
    SET VAR VPH TO PHONE IN #1
    SET VAR VADR TO ADDRESS IN #1
    SET VAR VCITY TO CITY IN #1
    SET VAR VST TO STATE IN #1
    SET VAR VZIP TO ZIPCODE IN #1
    NEWPAGE
    *(\ variable, form with values from matching record
    SET NULL *(* display null values as blank )
DRAW VEND WITH ALL
SET NULL -0-
WRITE ENTER TO TAB
AT 1
*( edit all fields )
EDIT VAR RETURN ESC PGDN
IF #RETURN = PGDN THEN *( save changes if PGDN pressed )
    CHANGE DEVACT TO .VACT IN #1
    CHANGE DEVLONG TO .VFULL IN #1
    CHANGE CONTACT TO .VPOC IN #1
    CHANGE PHONE TO .VPH IN #1
    CHANGE ADDRESS TO .VADR IN #1
    CHANGE CITY TO .VCITY IN #1
    CHANGE STATE TO .VST IN #1
    CHANGE ZIPCODE TO .VZIP IN #1
ENDIF
ENDIF
RETURN
*( =====
PROGRAM: S33PICK.CMD
AUTHOR : R. C. CAVALLARO
DATE WRITTEN: 7 JANUARY 1986
PURPOSE: DELETE A DEVELOPMENT ACTIVITY.

TABLES USED: VENDOR

```

VARIABLE FORMS USED: VEND

```
=====
FILLIN CH1 USING "ENTER DEVELOPMENT ACTIVITY ABBREVIATION: "
SET P1#1 FOR VENDOR WHERE DEVACT = .CH1
IF P1#1 THEN *(
  WRITE "THIS DEVELOPMENT ACTIVITY IS NOT ON FILE "
  WRITE "IT CANNOT BE DELETED"
  WRITE "PRESS ANY KEY TO CONTINUE . . "
  PAUSE
ELSE
  SET VAR VACT TO DEVACT IN #1
  SET VAR VFULL TO DEVLONG IN #1
  SET VAR VPOC TO CONTACT IN #1
  SET VAR VP#1 TO PHONE IN #1
  SET VAR VADR TO ADDRESS IN #1
  SET VAR VCITY TO CITY IN #1
  SET VAR VS#1 TO STATE IN #1
  SET VAR VZIP TO ZIPCODE IN #1
  NEWPAGE
  SET NULL " " *(
    DRAW VEND WITH ALL
    SET NULL -0-*(
      WRITE PGDN TO DELETE RECORD AT 1#1
      WRITE "ESC TO QUIT AT 1#1 60
      EDIT VAR VACT RETURN PGDN ESC
      IF #RETURN = PGDN THEN *(
        DELETE the record )
        NEWPAGE
        WRITE "ARE YOU SURE YOU WANT TO DELETE THIS"
        FILLIN ANS USING "RECORD (Y/N)"
        IF ANS = Y THEN
          DELETE ROWS FROM #1 *(
            after one last chance )
        ENDIF
      ENDIF
    RETURN
  *)
=====
```

```
=====
* (
  PROGRAM: S34PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 17 NOVEMBER 1985
  PURPOSE: PROMPTS USER FOR A SYSTEM ABBREVIATION AND THEN DISPLAYS
  ALL DOCUMENTS CREATED FOR THAT SYSTEM.

  TABLE USED: DOCLIST
=====
```

```

SET ERR VAR ERV "WHAT IS THE SYSTEM NAME"
FILLIN CH1 USING "WHAT IS THE SYSTEM NAME" *
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
SELECT DOCABR DOCNAME=30 DEVACT DOCDATE FROM DOCLIST WHERE SYSSNAME = .CH1
IF ERV NE O THEN
  WRITE "THERE ARE NO SYSTEMS BY THAT NAME ON FILE."
  WRITE "CHECK THE SPELLING AND TRY AGAIN."
  WRITE "COULD USE AN ASTERISK AS A WILD CARD IF YOU ARE UNCERTAIN."
  WRITE "OF THE SPELLING. FOR EXAMPLE USE JUMPS* TO FIND ANY NAME."
  WRITE "IN THE DATABASE THAT STARTS WITH JUMPS."
ENDIF
OUTPUT SCREEN
SET LINES 20
CLEAR CH1
SET VAR CHOICE = 0
RETURN
* ( reset loop control variable )

```

```

=====
* ( PROGRAM: S35PICK.CMD
  AUTHOR: R. C. CAVALLARO
  DATE WRITTEN: 29 NOVEMBER 1985
  PURPOSE: THIS PROGRAM PROVIDES VARIOUS INFORMATION PERTAINING TO THE
  MANAGEMENT OF SYSTEMS.
=====)

```

TABLE USED: SYSTEM

```

=====
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
WRITE "THESE SYSTEMS HAVE NO ANALYSTS ASSIGNED"
SELECT SYSSNAME IN SYSTEM SORT BY SYSSNAME WHERE ANALYST FAILS
IF IO NE Y THEN
  WRITE "PRESS ANY KEY TO CONTINUE . . . "
  PAUSE
ENDIF
WRITE " "
WRITE "THESE OPERATIONAL SYSTEMS ARE MISSING NUMBER OF PROGRAMS"
SELECT SYSSNAME IN SYSTEM SORT BY SYSSNAME WHERE PGMS FAILS AND CATEGORY = MCO
IF IO NE Y THEN

```

```
WRITE "PRESS ANY KEY TO CONTINUE . . . "
PAUSE
```

```
ENDIF
```

```
WRITE " "
```

```
WRITE "THESE OPERATIONAL SYSTEMS ARE MISSING LINES OF CODE"
SELECT SYSNAME IN SYSTEM SORT BY SYSNAME WHERE LOC FAILS AND CATEGORY = MCC
IF IO NE Y THEN
```

```
WRITE "PRESS ANY KEY TO CONTINUE . . . "
PAUSE
```

```
ENDIF
```

```
WRITE " "
```

```
WRITE "THESE SYSTEMS HAVE NO INDICATED PRIMARY USER"
SELECT SYSNAME IN SYSTEM SORT BY SYSNAME WHERE PRIUSER FAILS
OUTPUT SCREEN
SET LINES 20
RETURN
```

3

```
* ( =====
PROGRAM: S44PICK.CMD
AUTHOR : R.C.CAVALLARO
DATE WRITTEN: 17 NOVEMBER 1985
PURPOSE: PROMPTS USER FOR DOCUMENT TYPE AND THEN DISPLAYS ALL
SYSTEMS THAT HAVE HAD THAT TYPE DOCUMENT PREPARED.
```

TABLES USED: DOCLIST, DOCUMENT

```
===== ) * ( set_error_variable )
SET ERR VAR.ERV
FILLIN CH1 USING "WHAT IS THE DOCUMENT ABBREVIATION ( e.g RS ): "
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
SELECT SYSNAME DEVACT DOCNAME=35 DOCDATE FROM DOCLIST SORTED BY +
SYSNAME WHERE DOCABR CONTAINS .CH1
IF ERV NE O THEN
  WRITE "THERE ARE NO DOCUMENTS ON FILE FOR THAT ABBREVIATION."
  WRITE "PLEASE ENSURE THAT YOU SPELLED THE ABBREVIATION CORRECTLY."
  WRITE "PRESS L TO LIST ALL THE VALID ABBREVIATIONS ,"
  SET VAR.ANS TEXT
  FILLIN ANS USING "OR ANYTHING ELSE TO ABORT "
  IF ANS = L THEN
    SELECT DOCABR DOCTITLE DOCREF FROM DOCUMENT SORTED BY +
    LIFECA
```

```

ENDIF
CLEAR ANS
ENDIF
OUTPUT SCREEN
SET LINES 20
CLEAR CH1
SET VAR CHOICE = 0          *( reset loop control variable )

*( =====
PROGRAM: S64PICK.CMD
AUTHOR : R.CAVALLARO
DATE WRITTEN: 17 NOVEMBER 1985
PURPOSE: DISPLAYS ALL DOCUMENTS CREATED BY A GIVEN DEVELOPMENT
ACTIVITY AND THE SYSTEMS FOR WHICH THOSE DOCUMENTS WERE
CREATED.

TABLES USED: DOCLIST, VENDOR
=====)

SET ERR VAR ERV
FILLIN CH1 USING "ENTER DEVELOPMENT ACTIVITY ABBREVIATION ( e.g. DSA): "
NEWPAGE
IF IO = Y THEN
  OUTPUT SCREEN WITH PRINTER
  SET LINES 56
ENDIF
SELECT SYSNAME DOCABR DOCNAME=20 DOCDATE DEVACT FROM DOCLIST SORTED BY +
SYSNAME WHERE DEVACT CONTAINS .CH1
IF ERV NE "O" THEN
  WRITE "THERE ARE NO DEVELOPMENT ACTIVITIES ON FILE BY THIS NAME"
  SET VAR ANS TEXT
  WRITE "PRESS L FOR A LIST OF ALL VALID DEVELOPMENT ACTIVITIES, "
  FILLIN ANS USING "FOR ANYTHING ELSE TO ABORT"
  IF ANS = L THEN
    SELECT DEVACT DEVLONG FROM VENDOR SORTED BY DEVACT
  ENDIF
  CLEAR ANS
ENDIF
OUTPUT SCREEN
SET LINES 20
CLEAR CH1
SET VAR CHOICE = 0          *( reset loop control variable )
RETURN

```

COLUMN SYSTEMS MANAGEMENT MAIN MENU  
ADD  
CHANGE  
DELETE  
QUERY  
MANAGEMENT REPORTS  
RETURN TO MCASMIS MAIN MENU

SMENU01  
COLUMN SYSTEMS MANAGEMENT ADD MENU  
ADD A NEW SYSTEM  
ADD A NEW DOCUMENT  
ADD A NEW DEVELOPMENT ACTIVITY  
RETURN TO SYSTEMS MANAGEMENT MAIN MENU  
RETURN TO MCASMIS MAIN MENU

SMENU02  
COLUMN SYSTEMS MANAGEMENT EDIT MENU  
EDIT A SYSTEM  
EDIT A DOCUMENT FOR A SYSTEM  
EDIT DEVELOPMENT ACTIVITY INFORMATION  
RETURN TO SYSTEMS MANAGEMENT MAIN MENU  
RETURN TO MCASMIS MAIN MENU

SMENU03  
COLUMN SYSTEMS MANAGEMENT DELETE MENU  
DELETE A SYSTEM  
DELETE A DOCUMENT  
DELETE A DEVELOPMENT ACTIVITY  
RETURN TO SYSTEMS MANAGEMENT MAIN MENU  
RETURN TO MCASMIS MAIN MENU

SMENU04  
COLUMN SYSTEMS MANAGEMENT QUERY MENU  
CHANGE OUTPUT DEVICE  
CUSTOMIZED TALLY OF COLUMN  
SELECT DOCUMENTS BY SYSTEM  
SELECT SYSTEMS BY DOCUMENT TYPE  
LIST DATA DICTIONARY  
SELECT DOCUMENT & SYSTEM BY DEVELOPMENT ACTIVITY  
SELECT ALL INFORMATION ABOUT ONE SYSTEM  
RETURN TO SYSTEMS MANAGEMENT MAIN MENU  
RETURN TO MCASMIS MAIN MENU

SMENU05

ROW SYSTEMS MANAGEMENT MANAGEMENT REPORTS  
 CHG OUTPUT  
 DOCS DUE  
 TASK DOCS  
 PEND DOCS  
 PROJS  
 NEED DATA  
 LIST SYSTS  
 LIST DOCS  
 VENDS  
 ANALYSTS  
 SYSTEMS  
 OPER VIEW  
 DEVEL VIEW  
 HELP  
 RETURN  
 EXIT

```

*( ====== PROGRAM: SYSDRAW.CMD
      AUTHOR : R. C. CAVALLARO
      DATE WRITTEN: 21 NOVEMBER 1985
      PURPOSE: SETS ALL OF THE VARIABLE CONTENTS BEFORE DRAWING THE
              FORM SYS FOR EDITING VARIOUS DATA.
  
```

VARIABLE FORM USED: SYS

```

SET VAR VSYS TO SYSPNAME IN #1
SET VAR VLONG TO LNGLTITLE IN #1
SET VAR VCAT TO CATEGORY IN #1
SET VAR VANAL TO ANALYST IN #1
SET VAR VMENS TO MENS IN #1
SET VAR VI TO SDPI IN #1
SET VAR VII TO SDPILLI IN #1
SET VAR VIII TO SDPILLI IN #1
SET VAR VAMENS TO AMENS IN #1
SET VAR VAI TO ASDPPI IN #1
SET VAR VAI1 TO ASDPILLI IN #1
SET VAR VAI11 TO ASDPILLI IN #1
SET VAR VEXP TO EXPLIEE IN #1
SET VAR VRRI TO PRIUSER IN #1
SET VAR VENV TO ENV IN #1
SET VAR VCLASS TO CLASS IN #1
SET VAR VLANG TO COMPLANG IN #1
SET VAR VPRNO TO PGMS IN #1
SET VAR VLOC TO LOC IN #1
SET VAR VIMP TO IMPDATE IN #1
SET NULL
  
```

(\* null values displayed as blanks )

```
DRAW SYS WITH ALL  
SET NULL -O_ RETURN * ( redefine null value indicator )
```

```
* ( ====== PROGRAM: SYSLOAD.CMD  
      AUTHOR : R. C. CAVALLARO  
      DATE WRITTEN: 17 NOVEMBER 1985  
      PURPOSE: THIS PROGRAM ADDS A NEW SYSTEM TO THE DATABASE FORM  
                USING THE VARIABLE VALUES INPUT INTO THE VARIABLE FORM  
                SYS. ERROR CHECKING IS ENFORCED BY  
                THE RULES EXISTING IN THE DATA DICTIONARY.
```

```
TABLES USED: SYSTEM  
VARIABLE FORMS USED: SYS
```

```
===== WRITE " AT 1 1  
SET NULL -O-  
SET ERROR VAR ERV  
*( load the new record into the system table)  
LOAD SYSTEM  
  .V$SYS .VLONG .VCAT .VANAL .VMENS .VI VII VIII VEXP .VPRI .VENV .VCLASS +  
  .VLANG .VPRNO .VLOC .VIMP .VAMENS .VAI .VAI .VAI  
END  
IF ERV = O THEN  
  SET VAR OKAY = T  
  NEWPAGE  
  WRITE NEW RECORD ADDED FOR SYSTEM:  
  SHOW VAR VSYS AT 1 31  
  ELSE  
    WRITE "PRESS ANY KEY TO CONTINUE . . ." AT 24 10  
    PAUSE  
    NEWPAGE  
  ENDIF  
  SET VAR CHOICE = O  
RETURN
```

- 
- A - LIST ALL TTC'S THAT WERE MODIFIED BY A GIVEN PROJECT.
  - B - LIST ALL PROJECTS THAT MODIFIED A PARTICULAR TTC.

```

*( ====== PROGRAM: TTCXREF CMD
      AUTHOR : R. CAVALLARO
      DATE WRITTEN: 7 JANUARY 1986
      PURPOSE: THIS PROGRAM PROVIDES THE CAPABILITY TO ASCERTAIN EITHER
                WHICH TTC'S WERE MODIFIED BY A PARTICULAR PROJECT OR WHICH
                PROJECTS MODIFIED A GIVEN TTC. IT IS USEFUL IN ATTEMPTS TO
                FIND OUT WHY PROBLEMS SURFACED SUDDENLY IN A GIVEN AREA.

      SCREEN DISPLAYED: TTCSCKRN
      TABLES USED: TTC
      ======)

SET VAR ANS = C
WHILE ANS NE A AND ANS NE B THEN
  NEWPAGE
  DISPLAY TTCSCKRN
  FILLIN ANS USING 'ENTER EITHER A or B: '
ENDWHILE

WRITE "
  IF ANS = A THEN
    SET VAR PRID TEXT
    FILLIN PRID USING "PLEASE ENTER MPI PROJECT ID: "
    NEWPAGE
    SELECT TTC FROM TTC SORTED BY TTC WHERE MPI-ID = .PRID
  ELSE
    SET VAR CODE TEXT
    FILLIN CODE USING "PLEASE ENTER THE TTC: "
    NEWPAGE
    SELECT MPI-ID FROM TTC SORTED BY MPI-ID WHERE TTC = .CODE
  ENDIF
  CLEAR ANS
  CLEAR PRID
  CLEAR CODE
RETURN

```

APPENDIX B  
DATA DICTIONARY

A. TABLES IN THE DATABASE

Table: REPORTS

Read Password: NO  
Modify Password: NO

Column definitions

| # | Name  | Type | Length         | Key |
|---|-------|------|----------------|-----|
| 1 | RNAME | TEXT | 8 characters   | yes |
| 2 | RDATA | TEXT | 132 characters |     |

Current number of rows: 93

=====

Table: DOCUMENT

Read Password: NO  
Modify Password: NO

Column definitions

| # | Name     | Type | Length        | Key |
|---|----------|------|---------------|-----|
| 1 | DOCABR   | TEXT | 13 characters |     |
| 2 | DOCTITLE | TEXT | 45 characters |     |
| 3 | DOCREF   | TEXT | 12 characters |     |
| 4 | LIFECAT  | TEXT | 2 characters  |     |

Current number of rows: 33

=====

Table: DOCLIST

Read Password: NO  
Modify Password: NO

Column definitions

| # | Name    | Type | Length         | Key |
|---|---------|------|----------------|-----|
| 1 | SYSNAME | TEXT | 11 characters  | yes |
| 2 | DOCABR  | TEXT | 13 characters  |     |
| 3 | DOCNAME | TEXT | 109 characters |     |
| 4 | DEVACT  | TEXT | 16 characters  |     |
| 5 | DOCDATE | DATE | 1 value(s)     |     |
| 6 | DSTAT   | TEXT | 1 characters   |     |

Current number of rows: 234

=====

Table: SYSTEM

Read Password: NO  
Modify Password: NO

Column definitions

| # | Name     | Type | Length        | Key |
|---|----------|------|---------------|-----|
| 1 | SYSNAME  | TEXT | 11 characters | yes |
| 2 | LNGTITLE | TEXT | 72 characters |     |
| 3 | CATEGORY | TEXT | 3 characters  |     |
| 4 | ANALYST  | TEXT | 10 characters |     |
| 5 | MENS     | DATE | 1 value(s)    |     |
| 6 | SDPI     | DATE | 1 value(s)    |     |
| 7 | SDPII    | DATE | 1 value(s)    |     |

|    |          |         |    |            |
|----|----------|---------|----|------------|
| 8  | SDPIII   | DATE    | 1  | value(s)   |
| 9  | EXPLIFE  | TEXT    | 3  | characters |
| 10 | PRIUSER  | TEXT    | 9  | characters |
| 11 | ENV      | TEXT    | 10 | characters |
| 12 | CLASS    | TEXT    | 2  | characters |
| 13 | COMPLANG | TEXT    | 10 | characters |
| 14 | PGMS     | INTEGER | 1  | value(s)   |
| 15 | LOC      | INTEGER | 1  | value(s)   |
| 16 | IMPMDATE | DATE    | 1  | value(s)   |
| 17 | AMENS    | DATE    | 1  | value(s)   |
| 18 | ASDPI    | DATE    | 1  | value(s)   |
| 19 | ASDPII   | DATE    | 1  | value(s)   |
| 20 | ASDPIII  | DATE    | 1  | value(s)   |

Current number of rows: 102

---

Table: VENDOR  
 Read Password: NO  
 Modify Password: NO

Column definitions

| # | Name    | Type | Length        | Key |
|---|---------|------|---------------|-----|
| 1 | DEVACT  | TEXT | 16 characters | yes |
| 2 | DEVLONG | TEXT | 54 characters |     |
| 3 | CONTACT | TEXT | 20 characters |     |
| 4 | PHONE   | TEXT | 13 characters |     |
| 5 | ADDRESS | TEXT | 20 characters |     |
| 6 | CITY    | TEXT | 15 characters |     |
| 7 | STATE   | TEXT | 2 characters  |     |
| 8 | ZIPCODE | TEXT | 5 characters  |     |

Current number of rows: 39

---

Table: FORMS  
 Read Password: NO  
 Modify Password: NO

Column definitions

| # | Name  | Type | Length        | Key |
|---|-------|------|---------------|-----|
| 1 | FNAME | TEXT | 8 characters  | yes |
| 2 | FDATA | TEXT | 80 characters |     |

Current number of rows: 161

---

Table: RULES  
 Read Password: NO  
 Modify Password: NO

Column definitions

| # | Name     | Type    | Length        | Key |
|---|----------|---------|---------------|-----|
| 1 | NUMRULE  | INTEGER | 1 value(s)    |     |
| 2 | AND/OR   | TEXT    | 4 characters  |     |
| 3 | COLNAME1 | TEXT    | 8 characters  |     |
| 4 | TABLE1   | TEXT    | 8 characters  |     |
| 5 | BOOLEAN  | TEXT    | 4 characters  |     |
| 6 | COLNAME2 | TEXT    | 8 characters  |     |
| 7 | TABLE2   | TEXT    | 8 characters  |     |
| 8 | RULVALUE | TEXT    | 40 characters |     |

Current number of rows: 52

---

Table: PROJECT

Read Password: NO  
Modify Password: NO

Column definitions

| #  | Name     | Type    | Length        | Key |
|----|----------|---------|---------------|-----|
| 1  | MPI-ID   | TEXT    | 10 characters | yes |
| 2  | ACTIV-ID | TEXT    | 10 characters |     |
| 3  | PROJ-ID  | TEXT    | 3 characters  |     |
| 4  | SYSNAME  | TEXT    | 11 characters |     |
| 5  | CAT      | TEXT    | 1 characters  |     |
| 6  | PRIORITY | INTEGER | 1 value(s)    |     |
| 7  | FUNCPO   | TEXT    | 15 characters |     |
| 8  | ACTIVPO  | TEXT    | 15 characters |     |
| 9  | STATUS   | TEXT    | 1 characters  |     |
| 10 | DEVACT   | TEXT    | 16 characters |     |
| 11 | PROJNAME | TEXT    | 79 characters |     |
| 12 | STRUCW/T | DATE    | 1 value(s)    |     |
| 13 | DATEREQ  | DATE    | 1 value(s)    |     |
| 14 | REODUE   | DATE    | 1 value(s)    |     |
| 15 | SCHEDDUE | DATE    | 1 value(s)    |     |
| 16 | COMPLETE | DATE    | 1 value(s)    |     |
| 17 | TESTCOND | INTEGER | 1 value(s)    |     |
| 18 | TESTED   | INTEGER | 1 value(s)    |     |

Current number of rows: 14

=====  
Table: TTC  
Read Password: NO  
Modify Password: NO

Column definitions

| # | Name   | Type | Length        | Key |
|---|--------|------|---------------|-----|
| 1 | MPI-ID | TEXT | 10 characters |     |
| 2 | TTC    | TEXT | 3 characters  |     |

Current number of rows: 12

=====  
Table: DELIVER  
Read Password: NO  
Modify Password: NO

Column definitions

| # | Name    | Type | Length        | Key |
|---|---------|------|---------------|-----|
| 1 | MPI-ID  | TEXT | 10 characters |     |
| 2 | DELIVER | TEXT | 79 characters |     |

Current number of rows: 20

=====  
Table: DOCHIST  
Read Password: NO  
Modify Password: NO

Column definitions

| # | Name    | Type | Length         | Key |
|---|---------|------|----------------|-----|
| 1 | SYSNAME | TEXT | 11 characters  |     |
| 2 | DOCABR  | TEXT | 13 characters  |     |
| 3 | DOCNAME | TEXT | 109 characters |     |
| 4 | DEVACT  | TEXT | 16 characters  |     |
| 5 | DOCDATE | DATE | 1 value(s)     |     |
| 6 | DSTAT   | TEXT | 1 characters   |     |

Current number of rows: 0

Table: PROJHIST  
Read Password: NO  
Modify Password: NO

Column definitions

| #  | Name     | Type    | Length        | Key |
|----|----------|---------|---------------|-----|
| 1  | MPI-ID   | TEXT    | 10 characters |     |
| 2  | ACTIV-ID | TEXT    | 10 characters |     |
| 3  | PROJ-ID  | TEXT    | 3 characters  |     |
| 4  | SYSNAME  | TEXT    | 11 characters |     |
| 5  | CAT      | TEXT    | 1 characters  |     |
| 6  | PRIORITY | INTEGER | 1 value(s)    |     |
| 7  | FUNCPO   | TEXT    | 15 characters |     |
| 8  | ACTIVPO  | TEXT    | 15 characters |     |
| 9  | STATUS   | TEXT    | 1 characters  |     |
| 10 | DEVACT   | TEXT    | 16 characters |     |
| 11 | PROJNAME | TEXT    | 79 characters |     |
| 12 | STRUCW/T | DATE    | 1 value(s)    |     |
| 13 | DATEREQ  | DATE    | 1 value(s)    |     |
| 14 | REQDUE   | DATE    | 1 value(s)    |     |
| 15 | SCHEDDUE | DATE    | 1 value(s)    |     |
| 16 | COMPLETE | DATE    | 1 value(s)    |     |
| 17 | TESTCOND | INTEGER | 1 value(s)    |     |
| 18 | TESTED   | INTEGER | 1 value(s)    |     |

Current number of rows: 0

=====  
Table: TTCHIST  
Read Password: NO  
Modify Password: NO

Column definitions

| # | Name   | Type | Length        | Key |
|---|--------|------|---------------|-----|
| 1 | MPI-ID | TEXT | 10 characters |     |
| 2 | TTC    | TEXT | 3 characters  |     |

Current number of rows: 0

=====  
Table: DLVRHIST  
Read Password: NO  
Modify Password: NO

Column definitions

| # | Name    | Type | Length        | Key |
|---|---------|------|---------------|-----|
| 1 | MPI-ID  | TEXT | 10 characters |     |
| 2 | DELIVER | TEXT | 79 characters |     |

Current number of rows: 0

B. COLUMNS IN ALPHABETICAL ORDER

Column definitions

| Name     | Type | Length        | Table    | Key |
|----------|------|---------------|----------|-----|
| ACTIV-ID | TEXT | 10 characters | PROJECT  |     |
|          |      |               | PROJHIST |     |
| ACTIVPO  | TEXT | 15 characters | PROJHIST |     |
|          |      |               | PROJECT  |     |

|          |      |                |            |
|----------|------|----------------|------------|
| ADDRESS  | TEXT | 20 characters  | VENDOR     |
| AMENS    | DATE | 1 value(s)     | SYSTEM     |
| ANALYST  | TEXT | 10 characters  | SYSTEM     |
| AND/OR   | TEXT | 4 characters   | RULES      |
| ASDPI    | DATE | 1 value(s)     | SYSTEM     |
| ASDPII   | DATE | 1 value(s)     | SYSTEM     |
| ASDPIII  | DATE | 1 value(s)     | SYSTEM     |
| BOOLEAN  | TEXT | 4 characters   | RULES      |
| CAT      | TEXT | 1 characters   | PROJECT    |
|          |      |                | PROJHIST   |
| CATEGORY | TEXT | 3 characters   | SYSTEM     |
| CITY     | TEXT | 15 characters  | VENDOR     |
| CLASS    | TEXT | 2 characters   | SYSTEM     |
| COLNAME1 | TEXT | 8 characters   | RULES      |
| COLNAME2 | TEXT | 8 characters   | RULES      |
| COMPLANG | TEXT | 10 characters  | SYSTEM     |
| COMPLETE | DATE | 1 value(s)     | PROJHIST   |
|          |      |                | PROJECT    |
| CONTACT  | TEXT | 20 characters  | VENDOR     |
| DATEREQ  | DATE | 1 value(s)     | PROJHIST   |
|          |      |                | PROJECT    |
| DELIVER  | TEXT | 79 characters  | DELIVER    |
|          |      |                | DLVRHIST   |
| DEVACT   | TEXT | 16 characters  | DOCLIST    |
|          |      |                | VENDOR yes |
|          |      |                | PROJECT    |
|          |      |                | PROJHIST   |
|          |      |                | DOCHIST    |
| DEVLONG  | TEXT | 54 characters  | VENDOR     |
| DOCABR   | TEXT | 13 characters  | DOCHIST    |
|          |      |                | DOCLIST    |
|          |      |                | DOCUMENT   |
| DOCDATE  | DATE | 1 value(s)     | DOCHIST    |
|          |      |                | DOCLIST    |
| DOCNAME  | TEXT | 109 characters | DOCLIST    |

|          |         |                | DOCHIST  |     |
|----------|---------|----------------|----------|-----|
| DOCREF   | TEXT    | 12 characters  | DOCUMENT |     |
| DOCTITLE | TEXT    | 45 characters  | DOCUMENT |     |
| DSTAT    | TEXT    | 1 characters   | DOCHIST  |     |
|          |         |                | DOCLIST  |     |
| ENV      | TEXT    | 10 characters  | SYSTEM   |     |
| EXPLIFE  | TEXT    | 3 characters   | SYSTEM   |     |
| FDATA    | TEXT    | 80 characters  | FORMS    |     |
| FNAME    | TEXT    | 8 characters   | FORMS    | yes |
| FUNCPO   | TEXT    | 15 characters  | PROJHIST |     |
|          |         |                | PROJECT  |     |
| IMPDATE  | DATE    | 1 value(s)     | SYSTEM   |     |
| LIFECAT  | TEXT    | 2 characters   | DOCUMENT |     |
| LNGTITLE | TEXT    | 72 characters  | SYSTEM   |     |
| LOC      | INTEGER | 1 value(s)     | SYSTEM   |     |
| MENS     | DATE    | 1 value(s)     | SYSTEM   |     |
| MPI-ID   | TEXT    | 10 characters  | DELIVER  |     |
|          |         |                | DLVRHIST |     |
|          |         |                | PROJHIST |     |
|          |         |                | TTC      |     |
|          |         |                | PROJECT  | yes |
|          |         |                | TTCHIST  |     |
| NUMRULE  | INTEGER | 1 value(s)     | RULES    |     |
| PGMS     | INTEGER | 1 value(s)     | SYSTEM   |     |
| PHONE    | TEXT    | 13 characters  | VENDOR   |     |
| PRIORITY | INTEGER | 1 value(s)     | PROJECT  |     |
|          |         |                | PROJHIST |     |
| PRIUSER  | TEXT    | 9 characters   | SYSTEM   |     |
| PROJ-ID  | TEXT    | 3 characters   | PROJECT  |     |
|          |         |                | PROJHIST |     |
| PROJNAME | TEXT    | 79 characters  | PROJECT  |     |
|          |         |                | PROJHIST |     |
| RDATA    | TEXT    | 132 characters | REPORTS  |     |
| REQDUE   | DATE    | 1 value(s)     | PROJHIST |     |
|          |         |                | PROJECT  |     |

|          |         |               |          |     |
|----------|---------|---------------|----------|-----|
| RNAME    | TEXT    | 8 characters  | REPORTS  | yes |
| RULVALUE | TEXT    | 40 characters | RULES    |     |
| SCHEDDUE | DATE    | 1 value( s )  | PROJHIST |     |
|          |         |               | PROJECT  |     |
| SDPI     | DATE    | 1 value( s )  | SYSTEM   |     |
| SDPII    | DATE    | 1 value( s )  | SYSTEM   |     |
| SDPIII   | DATE    | 1 value( s )  | SYSTEM   |     |
| STATE    | TEXT    | 2 characters  | VENDOR   |     |
| STATUS   | TEXT    | 1 characters  | PROJECT  |     |
|          |         |               | PROJHIST |     |
| STRUCW/T | DATE    | 1 value( s )  | PROJECT  |     |
|          |         |               | PROJHIST |     |
| SYSNAME  | TEXT    | 11 characters | DOCLIST  | yes |
|          |         |               | SYSTEM   | yes |
|          |         |               | PROJECT  |     |
|          |         |               | PROJHIST |     |
|          |         |               | DOCHIST  |     |
| TABLE1   | TEXT    | 8 characters  | RULES    |     |
| TABLE2   | TEXT    | 8 characters  | RULES    |     |
| TESTCOND | INTEGER | 1 value( s )  | PROJHIST |     |
|          |         |               | PROJECT  |     |
| TESTED   | INTEGER | 1 value( s )  | PROJECT  |     |
| TESTED   | INTEGER | 1 value( s )  | PROJHIST |     |
| TTC      | TEXT    | 3 characters  | TTC      |     |
|          |         |               | TTCHIST  |     |
| ZIPCODE  | TEXT    | 5 characters  | VENDOR   |     |

#### C. COLUMN DESCRIPTIONS

- ACTIV-ID The internal document control number used by each activity to identify a given project originated by an MPI-ID.
- ACTIVPO The name of the activity project officer with primary responsibility for this project.
- ADDRESS The mailing address of the development activity.
- AMENS The ACTUAL date the MENS was completed.

ANALYST The name of the MPI-40 analyst who has responsibility for this system.

AND/OR An R:base column name used for data edit/entry checking in conjunction with the RULES function.

ASDPI The ACTUAL date that the System Decision Paper One was completed.

ASDPII The ACTUAL date that the System Decision Paper Two was completed.

ASDPIII The ACTUAL date that the System Decision Paper Three was completed.

BOOLEAN Used by the R:base RULES function for data editing/entering.

CAT The project category which may be (P)roduction mod, (T)est cycle, or (S)tandalone.

CATEGORY An abbreviation denoting whether the system is Marine Corps or DOD/DON and whether it is operational or being developed.

CITY The city in which the development activity is located.

CLASS The class designation of the system according to MCO P5231.1.

COLNAME1 Used by the R:base RULES function for data editing/entering.

COLNAME2 Used by the R:base Rules function for data editing/entering.

COMPLANG The programming language(s) in which the system is written,

COMPLETE The date that the project was completed. As such, this date also indicates the date that the project was closed.

CONTACT Point of contact in that development activity.

DATEREQ The date of the MPI request that originated the project.

DELIVER The deliverable(s) contracted to be received in conjunction with a given project.

DEVACT The development activity abbreviation.

DEVLONG The long title of the development activity.

DOCABR The document(s) abbreviation. Multiple documents are indicated by a slash between the abbreviations (e.g. UM/MM/OM).

DOCDATE The date of the life cycle document.

DOCNAME The title of the life cycle document.

DOCREF The reference for the life cycle document.

DOCTITLE The long title of the life cycle document type (e.g. RS = Requirements Statement).

DSTAT The status indicator of the document indicating whether the document is completed or pending.

ENV The environment in which the system processes.

EXPLIFE The expected life of the automated system.

FDATA Used by the R:base FORMS function in defining the formatted forms for data entry and editing.

FNAME Used by the R:base FORMS function in defining the formatted forms for data entry and editing.

FUNCPO The functional analyst assigned as the cognizant officer for the project.

IMPMDATE The date that the system was implemented.

LIFECAT Indicates whether the life cycle document type is a development item or a configuration item.

LNGTITLE The long title of the system.

LOC The number of thousands of lines of code in the system.

MENS The SCHEDULED due date for the MENS document.

MPI-ID The control number of the MPI source document that originated a project. This is the key field for all projects.

NUMRULE Used by the R:base RULES function for data editing/entering.

PROJ-ID The alphanumeric identifier assigned at the installation to which the project request was forwarded.

PGMS The number of computer programs that comprise the system.

PHONE The phone number of the development activity.

PRIORITY The priority assigned by the functional manager for that project

PRIUSER The primary user of the system.

PROJNAME A short description of the purpose of the project.

RDATA Used by the R:base REPORTS function.

REQDUE The requested due date specified in the project request.

RNAME Used by the R:base REPORTS function.

RULVALUE Used by the R:base REPORTS function.

SCHEDDUE The scheduled due date as negotiated with the development activity.

SDPI The SCHEDULED due date of the System Decision Paper One.

SDPII The SCHEDULED due date of the System Decision Paper Two.

SDPIII The SCHEDULED due date of the System Decision Paper Three.

SORTNBR Used by the R:base REPORTS function to print the systems properly sorted by category.

STATE The state in which the development activity is located.

STATUS The status of the project which is either (O)pen or (C)losed.

STRUCW/T The date that the structured walk through was conducted by the MPI functional analysts.

SYSNAME The system name abbreviation.

TABLE1 Used by the R:base RULES function for data editing/entering.

TABLE2 Used by the R:base RULES function for data editing/entering.

TESTCOND The total number of test conditions identified for satisfactory testing of the project.

TESTED The number of test conditions tested to date.

TTC The type transaction code which must be modified, deleted or created by the project.

ZIPCODE The zip code of the development activity.

#### D. ERROR MESSAGES

##### D01-THIS SYSTEM IS NOT ON FILE

Probable Cause: You tried to add or change a document specifying a system name that does not exist in the database.

Corrective Action: You must first add this system name using option 1 of the Systems Add Menu.

##### D02-DOC ABBREVIATION IS A MANDATORY FIELD

Probable Cause: You tried to add or change a document without specifying a document abbreviation.

Corrective Action: You must enter a value for this field.

##### D03-INVALID DOCUMENT ABBREVIATION

Probable Cause: You tried to add or change a document while specifying an invalid document abbreviation. Valid document abbreviations are contained in MPI Document number DOC-MPR-AIS. If one document combines two or more document types, the document abbreviations must be separated by a slash. For example, UM/MM/OM.

Corrective Action: Reinput after verifying according to above rules. Contact the data base administrator if a new document requirement causes the above rules to be unattainable.

##### D04-DEVEL ACTIVITY IS A MANDATORY FIELD

Probable Cause: You tried to add or change a document without specifying a development activity. Documents should only be on file after they have been assigned to a development activity for action. Therefore, this field may not be left blank.

Corrective Action: Specify a development activity.

##### D05-INVALID DEVELOPMENT ACTIVITY

Probable Cause: You tried to add or change a development activity that was not in the database. You must first add this development activity using option 2 from the Systems Management Add Menu. If more than one development activity is responsible for the document, the activity abbreviations must be separated with a slash. For example, DSA/IDEA.

Corrective Action: Verify and reinput.

##### D06-STATUS IS A MANDATORY FIELD

**Probable Cause:** You tried to add or change a life cycle document without specifying the status. The status must be entered.

**Corrective Action:** Enter a value for the status.

**D07-INVALID STATUS, MUST BE P OR C**

**Probable Cause:** You entered an invalid status for the life cycle document. All documents must either be pending or completed.

**Corrective Action:** Enter a status of either P or C.

**P01-MPI ID IS A MANDATORY FIELD**

**Probable Cause:** You tried to add or change a project without specifying a value for MPI-ID. This field is a key for the database and may not be left blank.

**Corrective Action:** Enter an MPI-ID.

**P02-SYSTEM NAME IS A MANDATORY FIELD**

**Probable Cause:** You tried to add or change a project without specifying the system name. Each project must be associated with a system and this field may not be left blank.

**Corrective Action:** Enter a system name.

**P03-THIS SYSTEM IS NOT ON FILE**

**Probable Cause:** You tried to add or change a project specifying a system name that does not exist in the database.

**Corrective Action:** You must first add this system name using option 1 of the Systems Add Menu.

**P04-STATUS MUST BE O OR C**

**Probable Cause:** You entered an invalid status for the project. The status must be either O (for open) or C (for closed).

**Corrective Action:** Enter a valid status.

**P05-CATEGORY MUST BE P, S OR T**

**Probable Cause:** You entered an invalid category for the project. The category must be P (for production mod), S (for stand alone) or T (for Test Cycle modification).

**Corrective Action:** Enter a valid category.

**P06-DEVEL ACTIVITY IS A MANDATORY FIELD**

**Probable Cause:** You tried to add or change a project without specifying a development activity. Since every project is opened for a specific activity, this field may not be left blank.

**Corrective Action:** Enter a development activity.

**P07-INVALID DEVELOPMENT ACTIVITY**

**Probable Cause:** You tried to add or change a development activity that was not on file. You must first add this activity using option 2 from the Systems Management Add Menu. If more than one activity is responsible for the project, the activity abbreviations must be separated with a slash. For example, HQMC/MCCDPA, KCMO.

**Corrective Action:** Verify and reinput.

**P08-INVALID NUMBER TESTED**

**Probable Cause:** You tried to input a number of test conditions that have been tested that was either more than the total number of test conditions, or when the total number of test conditions has not yet been input.

**Corrective Action:** Verify and reinput.

**S01-SYSTEM NAME IS A MANDATORY FIELD**

Probable Cause: You tried to add or change a system without specifying a system abbreviation. This value is a key field for the database and may not be left blank.

Corrective Action: Enter a system abbreviation.

**S02-SYSTEM NAME MUST BE UNIQUE**

Probable Cause: You tried to add or change a system abbreviation that already is defined for another system. The system abbreviation must be a unique value. Systems that are undergoing major enhancements should have an asterisk appended to the system name to make the name unique. For example, ARMS and ARMS\* are two existing systems in the database.

Corrective Action: Enter a unique system abbreviation.

**S03-INVALID CATEGORY**

Probable Cause: You tried to add or change a system with an invalid category. Valid categories are MCO, MCD, MCE, DNO or DNE. See the discussion of categories in this manual for more detail concerning categories.

Corrective Action: Input a valid category.

**S04-INVALID CLASS-MUST BE I,II OR -0-**

Probable Cause: You tried to add or change a system while specifying an invalid class. Class definitions are contained in MCO P5231.1. Valid classes are I, II or a null value if you are uncertain.

Corrective Action: Input a valid class.

**V01-DEVEL ACTIVITY IS A MANDATORY FIELD**

Probable Cause: You tried to add or change a development activity without specifying the abbreviation for that activity. This is a key field and may not be left blank.

Corrective Action: Enter an abbreviation for the development activity.

**V02-THIS ACTIVITY ALREADY EXISTS**

Probable Cause: You tried to add or change a development activity while specifying an abbreviation of an activity that already exists. The activity abbreviation must be unique.

Corrective Action: Input a unique development activity abbreviation.

APPENDIX C  
USER'S MANUAL

A. GETTING STARTED

1. System Requirements.

The Marine Corps Automated Systems Management Information System (MCASMIS) is designed to run in conjunction with the R:base 5000<sup>1</sup> Database Management System on the IBM PC, PC-XT, PC-AT and 100% compatible microcomputers.

Your system must have the following:

- DOS, version 2.0 or higher.
- 237K of main memory available after system configuration. A minimum of 320K is recommended.
- A color or monochrome monitor.
- A hard disk drive and one double-sided, double-density 5.25 inch floppy disk drive.

2. Overview.

MCASMIS is an integrated systems and project management information system designed to enhance the sponsorship of the many automated systems centrally managed at Headquarters, U.S. Marine Corps. Information concerning automated systems and projects that are ongoing concerning those systems is available in many different formats.

The following chapters will provide detailed explanations of all of the options available for each of the management functions.

In order to preserve the integrity of the database, there are certain functions which are restricted to the database administrator. These include archiving records to a history file, packing the database to recover space assigned to records that have been deleted, removing all

---

<sup>1</sup>R:base 5000 is a registered trademark of Microrim, Inc., Bellevue, Washington.

history records to a diskette to recover hard disk space, and reorganizing the database structure to accommodate future requirements. All such requests should be forwarded to the database administrator for action.

3. Special Considerations.

a. On-Line help.

The system was designed to be menu driven and should be self descriptive in most cases. From any of the menu screens (which are lists of options surrounded by a double-lined border), you may press F10 to get on-line instructions concerning that particular screen. Further, in many of the query options, an on-line display of valid arguments is offered if no records that match your response are found in the database.

b. Case distinction.

There is no distinction made between upper and lower case in any queries made against the database. For example, while searching for a system by the name of RMS, an input of rms, RMS, Rms, etc., would all work exactly the same. While entering or editing a record, however, you should remember that the data that is retrieved will be displayed in the case in which it was entered. If you enter a record into the database with a first name of john, then john will always be displayed when this record is selected, even though John or JOHN was the string on which you were searching.

c. Null values.

Within R:base, null values are stored as the symbol, -0-. This is the symbol that will be displayed when you edit a record, in order to distinguish it from a field that contains all blanks. The less experienced user might find the difference between a blank and a null value to be a bit nebulous. However, the distinction is indeed important in record selection. While entering data and receiving

reports, however, null values will be displayed as blanks. This approach was taken to give the output a more professional appearance. For the purpose of reports there really is no difference between null values and blanks.

d. Rules.

R:base allows rules to be defined from a global perspective which performs many of the editing functions. Any attempt to enter or change a record that violates any of these rules will be disallowed and you will be shown an appropriate error message and number on the screen. Refer to Appendix B for more information on each of these error numbers.

e. Date format.

The default format for all dates has been set to dd mmm yyyy. For convenience in inputting or editing data, you also have the option of entering dd/mm/yy if that format is more natural. That is, inputting a value of 1 Feb 1985 will result in exactly the same value being stored as when 1/2/85 is input. If, while attempting to input a date, R:base responds with an error message saying that a valid date is required, check the format and reenter according to the above guidelines.

f. Backup copies.

You are advised to make frequent backup copies of the database in order to assist in recovering from inadvertent loss of data due to machine or operator error.

4. Booting the System.

MCASMIS was designed to run on a hard disk. It was designed so that your R:base 5000 programs reside in a separate directory called RBASE. All of the MCASMIS programs should reside on a separate directory to facilitate backing up the programs and data. From this directory, type in the command,

MCASMIS

and the system will take over. Remember that this will only work properly if your directory with the R:base 5000 programs reside on a directory called RBASE. The directory with MCASMIS programs may have any name. Figure C.1 shows the MCASMIS Main Menu that is displayed upon first entering the system.

MCASMIS MAIN MENU  
(1) SYSTEMS MANAGEMENT  
(2) PROJECT MANAGEMENT  
(3) DATABASE MANAGEMENT  
(4) RETURN TO OPERATING SYSTEM

Figure C.1 MCASMIS Main Menu.

5. Exiting From R:base.

The final two options in most of the menu screens allow you to either return to the last menu or to return directly to the top menu. This last option was included in order to avoid having to page back through all of the preceding menus when you want to leave this application.

B. SYSTEMS MANAGEMENT

1. General.

If option 1 is selected from the MCASMIS Main Menu, you will be presented with the menu shown in Figure C.2. These options depict standard data manipulation functions that will route you to the appropriate menu for detailed selection as described in succeeding sections.

2. Adding Systems Records.

Figure C.3 shows the options that are available if you elect to choose the add option from the previous menu.

a. Adding a system.

When the need arises to add a new system into the existing database, you should select option 1 from the

```
SYSTEMS MANAGEMENT MAIN MENU
{1} ADD
{2} CHANGE
{3} DELETE
{4} QUERY
{5} MANAGEMENT REPORTS
{6} RETURN TO MCASMIS MAIN MENU
```

Figure C.2 Systems Management Main Menu.

```
SYSTEMS MANAGEMENT ADD MENU
{1} ADD A NEW SYSTEM
{2} ADD A NEW DOCUMENT
{3} ADD A NEW DEVELOPMENT ACTIVITY
{4} RETURN TO SYSTEMS MANAGEMENT MAIN MENU
{5} RETURN TO MCASMIS MAIN MENU
```

Figure C.3 Systems Management Add Menu.

Add Menu. A skeleton input form will be displayed on the screen and you will input the system abbreviation, system name, category and the name of the analyst who has primary responsibility for that system. Valid categories to be input and their meanings are:

- MCO - Marine Corps Operational System
- MCD - Marine Corps System under Development
- DNO - Operational System DON/DOD
- DND - DON/DOD System under Development

You will then be prompted to input additional items based on the category of that system. If the category was an operational system, you will be prompted for a different series of input than if the system was undergoing development. Any value not available should be left blank and a null value will be assigned to that field in the database.

The system name that is added must be UNIQUE. That is, that name should not already exist in the database. This is required because the system name is a key item that must point to a unique set of values that are functionally dependent on that system name. The Rules that are inherently defined in the data dictionary will preclude you from adding a non-unique system name, and it will provide an error message if you should attempt to do so.

Finally, you are given one last opportunity to abort from this add function by pressing the escape key (ESC), or to add the record by pressing the page down key (PGDN). Pressing the enter key (ENTER) allows you to tab to the next input area.

If you added a system, the name of the system added will be displayed and you will be asked if another system is to be added. This add process continues until you indicate that no more records are to be added by answering this prompt with anything other than "Y".

b. Adding a document.

Option 2 provides the capability to add a new life cycle document. This record should be added at the time that the task of preparing a specific document is assigned, with a status of P (for pending). You will be presented with a skeleton form to input values for the system name, the document abbreviation, the document name, the development activity, the document date and the status. If the document abbreviation satisfies more than one type, they should be separated by slashes. For example, a single document that serves as both a user's manual and a program maintenance manual should be input as UM/MM. Like the previous option, values unavailable for any field should be left blank and a null value will be stored in the record. The Enter key is used to tab to the next input area, PGDN to save the record and ESC to quit without saving and return to

the add menu. If the record is added, you are shown the name of the activity that was added and you are asked if another document is to be added. This process continues until you respond to this prompt with anything other than "Y".

c. Adding a development activity.

The first time that a new development activity is contracted to perform work, that activity should be added to the database. Selection of option 3 provides a skeleton input form that will prompt you for development activity abbreviation, full name of the activity, point of contact, phone number and full address. Blank fields will be stored as null values and the Enter, PGDN and ESC keys provide the same features as indicated in the other add options. You are allowed to input additional development activities until anything other than a "Y" is input to the continue prompt.

d. Finished adding.

Option 4 will return control to the Systems Management Main Menu.

e. Return to MCASMIS main menu.

Selection of the last option will bypass all intermediate menus and control is passed directly to the main menu.

3. Editing Systems Data.

Figure C.4 is the menu presented to you if the edit option is chosen from the main menu.

|                              |  |
|------------------------------|--|
| SYSTEMS MANAGEMENT EDIT MENU |  |
| { 1 }                        | EDIT A SYSTEM                          |
| { 2 }                        | EDIT A DOCUMENT FOR A SYSTEM           |
| { 3 }                        | EDIT DEVELOPMENT ACTIVITY INFORMATION  |
| { 4 }                        | RETURN TO SYSTEMS MANAGEMENT MAIN MENU |
| { 5 }                        | RETURN TO MCASMIS MAIN MENU            |

Figure C.4 Systems Management Edit Menu.

a. Editing a system.

You are prompted to enter the abbreviation for the system and a form is presented for editing. Use this option to change analyst name, category, assigned due dates for life cycle documents, or any other data element pertaining to the system. All current values for that record are displayed, and any of the values may be changed. Pressing the ENTER key TABS to the next input area, PGDN loads the record after changes are complete, and ESC quits the process and returns to the edit menu. Control is returned to the edit menu when you press ESC or have elected to load the edited system.

b. Editing a document.

When you select this option, you are prompted to enter the system and document abbreviation that is to be modified. A form is presented and any value may be changed by inputting the new value in the highlighted field and pressing PGDN to store it. A common use of this option would be to change the status from P to C when the document is delivered. Enter tabs to the next input area, PGUP finds the next match and ESC quits without editing. There may be multiple records in the file for a certain document type for one system, so you may have to page through the unwanted ones in order to get to the document in error.

c. Editing development activity information.

If a development activity record needs to be changed due to changes in address, name, point of contact, etc., option 3 should be selected and you will be prompted for the name of that activity. Any value in the record may be changed by typing over the old information in the highlighted area and pressing PGDN to save it. ENTER tabs to the next input area and ESC quits without modifying the record contents.

d. Finished editing.

Option 4 returns control to the Systems Management Main Menu.

e. Return to MCASMIS main menu.

Selection of this option bypasses all intermediate menus and control is passed directly to the main menu.

4. Deleting Systems Records.

Figure C.5 is the menu presented when option 3 is selected from the Systems Management Main Menu. In all of the following delete options, you are asked to verify that the record should be deleted before that deletion is executed. This ounce of prevention may seem cumbersome if you tire of giving commands more than once. However, the additional safety against inadvertently deleting a record by hitting a wrong key will more than compensate for this inconvenience.

|                                |  |
|--------------------------------|--|
| SYSTEMS MANAGEMENT DELETE MENU |  |
| {1}                            | DELETE A SYSTEM                        |
| {2}                            | DELETE A DOCUMENT                      |
| {3}                            | DELETE A DEVELOPMENT ACTIVITY          |
| {4}                            | RETURN TO SYSTEMS MANAGEMENT MAIN MENU |
| {5}                            | RETURN TO MCASMIS MAIN MENU            |

Figure C.5 Systems Management Delete Menu.

a. Deleting a system.

You are prompted for a system name when this option is selected. A form is presented which contains all current values for that record. You press PGDN to delete this record or ESC to quit the process without deleting. If any open projects are on file for the system, you are advised that deletion will not be allowed until those projects are either closed or deleted. However, it should be noted that documents pertaining to any system that is deleted must be deleted individually using option 2 of this menu.

b. Deleting a document.

You are prompted for a document and a system abbreviation and is then shown records that satisfy those criteria one at a time. You press PGDN to delete that document, PGUP to find the next matching document, or ESC to quit and return to the delete menu.

c. Deleting a development activity.

You are prompted for a development activity abbreviation and is then shown the fields for that selection on a labeled form. You again press PGDN to delete that record or ESC to quit without deleting. Control is returned to the delete menu when no matching record is found or ESC is pressed.

d. Finished deleting.

Option 4 returns control to the Systems Management Main Menu.

e. Return to MCASMIS main menu.

Selection of this option bypasses all intermediate menus and control is passed directly to the main menu.

5. The Systems Query Process.

Selecting option 4 from the main menu displays the menu shown in Figure C.6 to you. Queries are generally nonrecurring, ad hoc request that require you to first answer a prompt to delineate the search criteria. This subsystem is designed to answer the types of questions that are presented below.

a. Changing the designated output device.

Option one enables you to change the output device from the screen to printer. If this option is selected, the menu shown in Figure C.7 is presented for you to designate the output device. Once chosen, the output device remains in effect until the current menu is exited, or you again change the output designation using this option. There are two important features of this option as follows.

|       |  |
|-------|--|
|       | SYSTEMS MANAGEMENT QUERY MENU                    |
| { 1 } | CHANGE OUTPUT DEVICE                             |
| { 2 } | CUSTOMIZED TALLY BY COLUMN                       |
| { 3 } | SELECT DOCUMENTS BY SYSTEM                       |
| { 4 } | SELECT SYSTEMS BY DOCUMENT TYPE                  |
| { 5 } | LIST DATA DICTIONARY                             |
| { 6 } | SELECT DOCUMENT & SYSTEM BY DEVELOPMENT ACTIVITY |
| { 7 } | SELECT ALL INFORMATION ABOUT ONE SYSTEM          |
| { 8 } | RETURN TO SYSTEMS MANAGEMENT MAIN MENU           |
| { 9 } | RETURN TO MCASMIS MAIN MENU                      |

Figure C.6 Systems Management Query Menu.

When the printer is selected as the output device, output will be routed simultaneously to the screen also. This allows you the opportunity to manually adjust the paper form feed at the top of each new page of output before proceeding by pressing any key.

Not all of the queries and management reports may be routed directly to the printer. Those reports which disallow this designation (for example, those that use skeleton forms), are specified in the help screens for each subsystem. Having the output designation set to printer when choosing these restricted options will have no effect and the output will still be presented to the screen.

|        |                    |
|--------|--------------------|
| SCREEN | SELECT OUTPUT MENU |
|        | PRINTER            |

Figure C.7 Select Output Menu.

b. Customized tally by column.

This is one of the more flexible queries in the subsystem. It provides a count of the number of occurrences with each value in the database. For example, "tally analysts in system" would list the names of every analyst on

file and how many systems for which they had responsibility. Similarly, "tally devact in doclist" would search the doclist table and provide the names of every development activity and how many documents each had prepared. You are first presented with a list of all tables in the database. After choosing one of these tables, you are presented with a list of columns that comprise the table selected. Appendix B lists all tables, columns and column definitions. Although this option does require some familiarity with the database structure, a little practice using this command in conjunction with the data dictionary provided will reap great benefits in establishing valuable insight into the contents of the database. The possibilities are endless; for example: How many projects for each system could be obtained by, tally sysname in project. Or tally docabr in doclist would give information on how many of each type life cycle document existed on file.

c. How well is this system documented?

After selecting option 3, you are prompted for a system abbreviation. The document abbreviation, document title, development activity and document date are displayed for all documents that pertain to that system. The output includes documents that are currently being prepared as well as those that have been completed.

d. Where can I find a sample of a Functional Description?

To get an idea of what documents exist, and what development activities have experience in writing that particular document type, you select option 4 and are prompted for a document abbreviation. You are then presented with the system name, development activity, document title and document date for all documents that match

that abbreviation.

e. What are the database definitions?

Option 5 provides an on-line data dictionary that lists all column names with the tables names that contain that field, plus a listing of each individual table with format of the fields and whether or not that field is a key value. Key fields enable faster searching by indexing the values of those fields separately and the inexperienced user should not be concerned with this information. Although not essential, it is recommended that this option be designated to the printer output device.

f. What documents has this contractor authored before?

After selecting option 6, you are prompted to input an abbreviation for a development activity. You are then shown the system name, document abbreviation, document title, document date and development activity for all documents on file for which this activity authored or coauthored. The development activity is displayed because in many cases, the document was coauthored by more than one contractor and this fact should be known when evaluating this question.

g. Selecting all information about a system.

You are prompted for a system abbreviation and are shown all information pertaining to that system on a labeled form. After pressing any key, the query menu will be redisplayed.

h. Finished querying.

Option 8 returns control to the Systems Management Main Menu.

i. Return to operating system.

Option 9 bypasses all intermediate menus and control is returned directly to the operating system.

6. Systems Management Reports.

While the preceding options are often useful in generating answers to ad hoc requests, this option is really the backbone of the Systems Management subsystem. Figure C.8 depicts the sixteen options that are available to you after selecting the Management Reports Option from the Systems Management Main Menu.

|                    |            |            |            |
|--------------------|------------|------------|------------|
| SYSTEMS MANAGEMENT | MANAGEMENT | REPORTS    |            |
| CHG OUTPUT         | DOCS DUE   | TASK DOCS  | PEND PROJS |
| PEND DOCS          | NEED DATA  | LIST SYSTS | LIST DOCS  |
| LIST VENDS         | ANALYSTS   | SYSTEMS    | OPER VIEW  |
| DEVEL VIEW         | HELP       | RETURN     | EXIT       |

Figure C.8 Systems Management Management Reports.

Table X provides a list of the reports that are generated upon your request.

#### C. PROJECT MANAGEMENT

##### 1. General.

Selection of option 2 from the MCASMIS Main Menu will cause the menu shown in Figure C.9 to be presented. This menu should look very familiar since it is virtually identical to the initial options presented under Systems Management. The following sections provide detailed descriptions of each of these options.

##### 2. Adding Project Records.

Figure C.10 shows the options that are available if you select the add option from the previous menu.

###### a. Adding a new project.

To add a new project to the database, you select option 1 and are presented with a skeleton form in which to input various pieces of information.

The MPI-ID, system name and development activity fields are mandatory and may not be left blank. The system

TABLE X  
SYSTEMS MANAGEMENT REPORTS

|            |  |
|------------|--|
| CHG OUTPUT | Enables you to change the output device to which the selected reports are routed to screen or screen with printer.   |
| DOCS DUE   | These options prompt you for the number of days in the future on which to base the search, and display the system name and the scheduled date that the indicated document was due but as yet has not been completed. |
| TASK DOCS  | Provides a listing of the system names and the scheduled dates for all indicated documents that have assigned due dates, but have not yet been tasked to anyone to complete.   |
| PEND DOCS  | Lists all pending documents (not yet completed/delivered).   |
| PEND PROJS | Lists all open projects for a given system. You are prompted to enter a system name for this option.   |
| NEED DATA  | Checks the database and provides a listing of data items that are missing for operational systems.   |
| LIST SYSTS | Provides a listing of all automated systems in the database sorted by category and system name.  |
| LIST DOCS  | Provides a listing of all documents in the database sorted by system.  |
| LIST VENDS | Provides a listing of all development activities and their phone numbers in alphabetical order.  |
| ANALYSTS   | Provides a listing of analyst name, system name, category and long title sorted by analyst.  |
| SYSTEMS    | Provides the same listing as ANALYSTS except the sort order is by system name.   |
| OPER VIEW  | Provides an overview of Marine Corps operational systems.  |
| DEVEL VIEW | Provides an overview of Marine Corps enhancement and development systems.  |
| HELP       | Provides on-line help for the above management reports.  |
| RETURN     | Returns control to the Systems Management Main Menu.   |
| EXIT       | Leaves the MCASMIS System and returns control to the operating system after closing the database.  |

```
PROJECT MANAGEMENT MAIN MENU
(1) ADD
(2) CHANGE
(3) DELETE
(4) QUERY
(5) MANAGEMENT REPORTS
(6) RETURN TO MCASMIS MAIN MENU
```

Figure C.9 Project Management Main Menu.

```
PROJECT MANAGEMENT ADD MENU
(1) ADD A NEW PROJECT
(2) ADD TTC TO EXISTING PROJECT
(3) ADD DELIVERABLE TO EXISTING PROJECT
(4) RETURN TO PROJECT MANAGEMENT MAIN MENU
(5) RETURN TO MCASMIS MAIN MENU
```

Figure C.10 Project Management Add Menu.

name and development activity that are being added must already be in the database. If not, an error message will be displayed and you should use the appropriate option from the Systems Management subsystem to add this record.

The status field refers to whether the project is open or closed. Enter an O if the project is just being added, later using the change process to modify the status to C for closed.

Valid categories are P if the project entails a production modification to an existing application, an S if the project is a stand alone product, or a T if the project is included in a Test Cycle environment.

All other values may be left blank if the information is not available at the time the project is being opened.

After entering the applicable data, you press PGDN to load the new record or ESC to abort the load process. ENTER allows you to tab repeatedly to the next input area. If PGDN is pressed, the entered information is validated in accordance with the rules specified above. If the information is valid, the record is added and you are so informed. If not, an appropriate error message is displayed.

If the project being added pertains to either JUMPS/MMS or REMMPS, you are then prompted to enter the Type Transaction Codes (TTC) that are being modified, deleted or created by this project.

After adding any project, you are given the opportunity to add the names of one or more deliverables that are to be received as part of that project. The ENTER key loads the deliverable, and ESC aborts the process and returns to the Project Management Add Menu.

b. Adding a TTC to an existing project.

Option 2 provides the capability to add a TTC to an already existing project. Often times, after a project has been submitted, subsequent analysis of the problem or requirements uncover ramifications to additional TTCS than those that were originally identified. This option permits the addition of those TTCS and maintains the integrity of the database.

You are first prompted to enter the MPI project ID number of the existing project. If no such project is on file, an error message is displayed. If the project number is valid (i.e., it already exists on file), you are presented with a form to add as many TTCS as desired.

Control is returned to the Project Management Add Menu when you indicate that you do not desire to add a TTC to any more projects at the given prompt.

c. Adding a deliverable to an existing project.

Option 3 allows you to add a deliverable to an existing project. As in the option for adding a TTC to an existing project, you are prompted to enter the MPI project ID number and the validity of this number is first checked before prompting you to input the deliverable.

Control is returned to the Project Management Add Menu when you indicate that deliverables are not to be added to any more projects.

d. Finished adding.

Option 4 will return control to the Project Management Main Menu.

e. Return To MCASMIS main menu.

Selection of option 5 will bypass all intermediate menus and pass control directly to the main menu.

### 3. Editing Project Data.

If you select the change option from the main menu, Figure C.11 is presented.

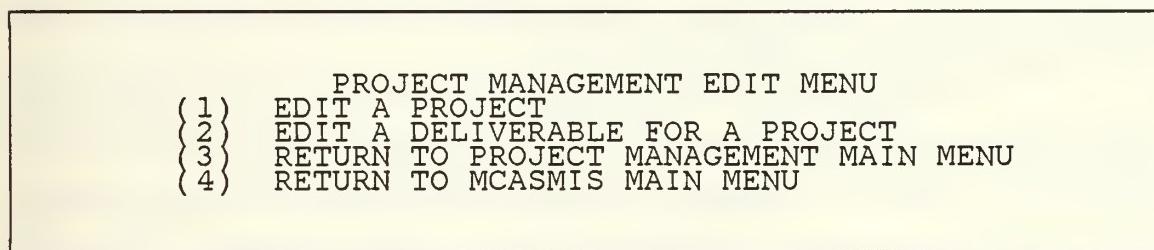


Figure C.11 Project Management Edit Menu.

a. Editing a project.

You are prompted to enter the MPI project identification number. If no such project exists, an error message is displayed. If valid, a form is presented with all values displayed for that project and you may edit any desired field except for the MPI ID field. ENTER tabs to the next input field, PGDN loads the record and you are informed of a successful load, and ESC aborts the edit process.

No priority may be assigned or changed to a value that is higher than the number of projects on file for that development activity that have an open status. The system will automatically assign it a priority of the highest number if the field is left blank or if an attempt is made to assign a priority that is too large.

Also, if the priority is modified during this edit process, all other priorities for open projects for that activity are automatically adjusted. After modifying a priority, get a listing of projects by priority (option 5 from the Project Management Query Menu) to see the results.

Control is returned to the Project Management Edit Menu when you elect to either load the modified record, or abort by pressing the ESC key.

b. Editing a deliverable for a project.

As in the previous option, you are prompted to enter an MPI project identification number and this input is validated. If valid, you are shown all deliverables on file for that project, one at a time, and are permitted to modify the description of that deliverable.

Since there may be more than one deliverable for any project, you may have to page through all of them before getting to the one desired. PGDN loads the modified record, PGUP finds the next deliverable for that project and ESC aborts the process and returns to the Project Management Edit Menu.

c. Finished editing.

Option 3 returns control to the Project Management Main Menu.

d. Return to the MCASMIS main menu.

Selection of this option bypasses all intermediate menus and returns control directly to the main menu. The built in RULES preclude editing data to an illegal value in all of the editing options.

4. Deleting Project Records.

Figure C.12 is the menu presented when option 3 is selected from the Project Management Main Menu. As in the Systems Management delete process, you are asked to verify the deletion before execution.

|                                |  |
|--------------------------------|--|
| PROJECT MANAGEMENT DELETE MENU |  |
| { 1 }                          | DELETE A SINGLE PROJECT                |
| { 2 }                          | DELETE ALL PROJECTS FOR ONE SYSTEM     |
| { 3 }                          | DELETE A TTC FOR A PROJECT             |
| { 4 }                          | DELETE A DELIVERABLE FOR A PROJECT     |
| { 5 }                          | RETURN TO PROJECT MANAGEMENT MAIN MENU |
| { 6 }                          | RETURN TO MCASMIS MAIN MENU            |

Figure C.12 Project Management Delete Menu.

a. Deleting a single project.

You are prompted to enter the MPI project identification number and an error message is displayed if the project is not on file. If the record is found, a form with all values for that project is displayed and you press either PGDN to delete the record or ESC to abort the delete process.

If PGDN is pressed, you are asked to verify that the delete is correct before proceeding.

When a project is deleted, all projects of lower priority are automatically adjusted. Also, all TTC and deliverable records on file for this project are automatically deleted when the project is deleted.

b. Deleting all projects for one system.

You are prompted to enter the system abbreviation and are given the opportunity to verify that the delete should be executed. Again, all priorities are automatically adjusted as a result of any projects deleted by this option. TTC and deliverable records for these projects are also automatically deleted.

c. Deleting a TTC for a project.

You are prompted to enter the MPI project identification number and are then shown all TTC records on file for that project one at a time. Pressing PGDN deletes the TTC, PGUP finds the next TTC record for that project and ESC aborts the process and returns control to the Project Management Delete Menu.

d. Deleting a deliverable for a project.

After answering the prompt for the MPI project identification number, you are shown all deliverable records one at a time and again press PGDN to delete the record, PGUP to find the next match or ESC to return control to the Project Management Delete Menu without deleting.

e. Finished deleting.

Option 5 returns control to the Project Management Main Menu.

f. Return to MCASMIS main menu.

Selection of this last option again bypasses all intermediate menus and returns control to the main menu.

5. The Project Query Process.

Selecting the query option from the Project Management Main Menu will display the menu shown in Figure C. 13.

|                               |  |
|-------------------------------|--|
| PROJECT MANAGEMENT QUERY MENU |  |
| (1)                           | CHANGE OUTPUT DEVICE                           |
| (2)                           | ALL OPEN PROJECTS SORTED BY SYSTEM             |
| (3)                           | ALL OPEN PROJECTS FOR ONE SYSTEM               |
| (4)                           | ALL OPEN PROJECTS SORTED BY SCHEDULED DUE DATE |
| (5)                           | ALL OPEN PROJECTS FOR AN ACTIVITY BY PRIORITY  |
| (6)                           | TTC CROSS REFERENCE MENU                       |
| (7)                           | ALL INFORMATION ON A PARTICULAR PROJECT        |
| (8)                           | RETURN TO PROJECT MANAGEMENT MAIN MENU         |
| (9)                           | RETURN TO MCASMIS MAIN MENU                    |

Figure C. 13 Project Management Query Menu.

The following types of questions are answered by this subsystem.

- a. Changing the output device.

This option works exactly the same as it does in the systems management subsystem.

- b. In which systems is work being performed?

This option provides the overall picture of all open projects sorted by the system in which the work is being performed, and the priority assigned to that project. The scheduled due date of each project is also included and you can see at a glance which projects and which systems require additional attention.

- c. What work is being done in my system?

This option provides the responsible analyst to get a listing of all open projects for that system sorted by priority.

- d. Which projects are scheduled to be completed next?

All open projects are listed sorted by the scheduled due date. Projects that are overdue can be seen at a glance and given immediate attention.

- e. What projects are being done at one activity?

You are prompted to input the activity abbreviation and are shown a listing of all open projects for that activity sorted by priority.

- f. Which project(s) modified TTC 020?

This option could be invaluable for tracking down the cause of a production problem. Also available with this option is the answer to, "what TTCs were modified by a given project". You elect which criterion to set.

- g. Selecting all information about a project.

You are prompted for the MPI project identification number and are then shown all information pertaining to that system on a labeled form. Any TTC or deliverable information connected with that project is also displayed.

h. Finished querying.

Option 8 returns control to the Project Management Main Menu.

i. Return to MCASMIS main menu.

Selection of this last option bypasses all intermediate menus and returns control directly to the main menu.

#### 6. Project Management Reports.

Figure C.14 depicts the menu presented when the management reports option is selected from the Project Management Main Menu.

|                                       |  |
|---------------------------------------|--|
| PROJECT MANAGEMENT MANAGEMENT REPORTS |  |
| {1}                                   | CHANGE OUTPUT DEVICE                               |
| {2}                                   | OPEN PROJECTS THAT ARE OVERDUE OR DUE SOON         |
| {3}                                   | PROJECTS BY FUNCTIONAL OR ACTIVITY PROJECT OFFICER |
| {4}                                   | LOG OF PROJECTS BY DATE OF MPI REQUEST             |
| {5}                                   | LOG OF DELIVERABLES FOR PROJECTS ON FILE           |
| {6}                                   | LOG OF PROJECTS BY SCHEDULED DUE DATE              |
| {7}                                   | TEST CYCLE STATUS REPORT                           |
| {8}                                   | RETURN TO PROJECT MANAGEMENT MAIN MENU             |
| {9}                                   | RETURN TO MCASMIS MAIN MENU                        |

Figure C.14 Project Management Management Reports.

Table XI provides a list of the reports that are generated at your request.

#### D. DATABASE MANAGEMENT

##### 1. General

Most of the functions of the Database Management subsystem should be restricted to the system manager located at Headquarters, U.S. Marine Corps. Only option 1 should be used by all other users of the system. Future versions of the MCASMIS system will incorporate security to ensure that unauthorized users of these options do not jeopardize the integrity of the database.

TABLE XI  
PROJECT MANAGEMENT REPORTS

- 1 - Enables you to change the output device to which the subsequent reports are routed.
- 2 - Provides a detailed report by system of all open projects including the priority and scheduled due date.
- 3 - Provides a list of projects under the cognizance of either a functional or activity project officer sorted by MPI ID number.
- 4 - Provides a detailed log of all MPI requests, with the most recent requests shown first.
- 5 - Provides a detailed listing of all deliverables on file, sorted by MPI project identification number.
- 6 - Provides a listing of all open projects, with the earliest scheduled due dates appearing first.
- 7 - Produces a test cycle status report sorted by activity project identifiers for use in tracking the quality control being performed by the development activity.

Unfortunately, because of the special configuration considerations involved with the distributed processing sites, this security cannot be added at the present, since these remote sites require the capabilities of option 1 to transmit their input data to HQMC for consolidation.

Figure C.15 shows the menu that is displayed to you when the database management option is selected from the main menu.

|                               |  |
|-------------------------------|--|
| DATABASE MANAGEMENT MAIN MENU |  |
| { 1 }                         | UNLOAD DATA TO SEND TO HQMC            |
| { 2 }                         | MERGE DATA FROM DISTRIBUTED SITES      |
| { 3 }                         | PACK DATABASE TO RECOVER UNUSED SPACE  |
| { 4 }                         | ARCHIVE RECORDS                        |
| { 5 }                         | REMOVE ALL ARCHIVED RECORDS TO HISTORY |
| { 6 }                         | RETURN TO MCASMIS MAIN MENU            |

Figure C.15 Database Management Main Menu.

## 2. Unloading Data to Send to HQMC

This is normally the only option that should ever be selected by any user other than the system manager at HQMC. It enables you to unload all data pertaining to projects, including associated TTC and deliverable records, for subsequent transmittal to HQMC for consolidation. It is envisioned that this file will be small enough to allow rapid transmission over standard voice telephone lines from micro to micro. While 2400 baud modems will provide faster response, the current 1200 baud modems should be sufficient to accomplish this data transfer.

Of course, depending on the urgency of the need for updated data, the diskette containing this data could even be mailed if the situation so dictated.

You should first have a pre formatted diskette that does not contain any data already on it. You are presented with the prompt for which drive will contain the data that is to be unloaded. Only drives A and B are valid for this purpose. After properly answering this prompt, all of the data is automatically unloaded and you are returned to the MCASMIS Main Menu.

It is very important that no additional data be added at the remote site until the new database diskette is returned from HQMC. This manual coordination is required to keep the data synchronized because of the unique peculiarities of the configuration.

When the new diskette is returned from HQMC, you should load that diskette into either drive A or B, and from the MCASMIS directory, issue the DOS copy command with that drive. For example,

COPY A:MCMIS\*.RBS

This command should be typed exactly as shown, including the asterisk, with only the drive designator allowed to change to be either A or B as appropriate.

### 3. Merging Data from Remote Sites at HOMC

The data that was unloaded by you at the remote site is in an ASCII file with embedded R:base 5000 commands that will allow merging this data into the MCASMIS database. If this data was transmitted over telephone lines, the receiver should give it a filename of "OUT#DATA". If the data diskette was received via mail, it will already have this filename.

The receiver simply places this diskette into either drive A or B, and answers the first question with that drive letter. A message is displayed telling you to standby while the data is input and all duplicate records are automatically deleted.

You should then insert an empty, formatted diskette into either drive A or B, and answer the next prompt for which drive should contain the data to be returned to the remote site. Then, the data is automatically reloaded, all space is recovered from the deleted records, and both the hard disk and the floppy diskette will contain the updated, complete database.

The floppy diskette will then be mailed to the remote site. You will use the DOS copy command to copy the new database at the remote site, and the system can then continue to be used. Control is returned to the Database Management Main Menu at the conclusion of this action.

### 4. Packing the Database

When records are deleted from the database, the space formerly occupied by those records is still not useable until the database is either packed or reloaded. Option 3 allows the system manager to recover this unuseable space. Drives A, B or C may be designated at the first prompt as the drive to be used for scratch files. After this prompt is correctly answered, every thing else is performed automatically by the MCASMIS system, and control is returned to the Database Management Main Menu.

## 5. Archiving Records

As the database gets too large, access times for all queries and other processing increases. Periodically, the system manager can select this option to archive those document and project records that are no longer current. For example, all development item documents for systems life cycle management need no longer be retained once the system has been implemented. Similarly, projects that have been completed can be archived periodically. Once archived, these records remain in the database in a different table, and will no longer be retrievable from the main MCASMIS system. They will, however, still be available to the system manager using the R:base programming language, or CLOUT.

After selecting this option, you are presented with another menu and given the opportunity to archive either documents or projects.

### a. Archiving Documents

This option will archive all development item documents for a system that has been implemented. You are prompted for the system abbreviation, and the MCASMIS system does everything else. Upon completion, control is returned to the previous menu to archive additional records, or return to the Database Management Main Menu.

### b. Archiving Projects

You are first prompted for a date. All projects that were completed prior to this date will be archived, and the rest will remain in the main MCASMIS system. It may be desireable to archive records by calendar year, by test cycle, monthly, or some other appropriate schedule.

While the archival of document records causes no problems, since all documents are input centrally at HQMC, project archival should be performed in close coordination with remote sites, since they also have to be archived at

those sites at the same time. Failure to do this will result in the projects returning to the database the next time data is transmitted to HQMC from those remote sites.

When a project record is archived, all associated TTC and deliverable data is also archived automatically. Only completed projects may be archived. Control is returned to the archival menu at the completion of this function.

#### 6. Removing Archived Records to History

Archiving records simply moves them to another table within the MCASMIS database in order to increase the responsiveness of the MCASMIS system. No space on the disk is recovered by that action.

As disk space becomes a concern, those records that have been archived can be removed to a floppy diskette for permanent retention using option 5 of the Database Management Main Menu.

You may designate either drive A or B for this purpose. After inserting an empty, formatted diskette into one of these drives, the prompt is answered and the system takes over. Upon completion, you must then execute either option 2 or 3 from the Database Management Main Menu before that space will be recovered.

#### E. SUGGESTIONS AND COMMENTS

MCASMIS is a prototype system. The primary purpose of this project is to publish a functional model of a system for user perusal. While the system has been carefully checked for syntactical and logic errors, it cannot be assumed to be completely validated. Any errors that are discovered may be brought to the attention of the author.

Similarly, comments and suggestions concerning additions and deletions to this system are encouraged. It is only through this iterative process that an optimal system can be developed. Please forward all such comments to:

MAJ R. C. Cavallaro

Manpower Information Systems Liaison Unit  
Code MPI-70  
Marine Corps Finance Center  
Kansas City, MO 64197-0001

#### LIST OF REFERENCES

1. Cash, James I., McFarlan, F. Warren, and McKenney, James L., Corporate Information Systems Management: Text and Cases, p. 35, Richard D. Irwin, Inc., 1983
2. Powers, Michael J., Adams, David R., and Mills, Harlan D., Computer Information Systems Development: Analysis and Design, p. 609, South-Western Publishing, 1984.

## BIBLIOGRAPHY

- Ackoff, Russell L. "Management Misinformation Systems," Management Science, Vol. 14, No. 4, December 1967.
- Alter, Steven, Decision Support Systems, Addison-Wesley, New York, 1980.
- Alter, Steven, "A Taxonomy of Decision Support Systems," Sloan Management Review, Vol. 19, No. 1, Fall 1981.
- Appleton, D. S. "Data Driven Prototyping," Datamation, Vol. 29, November 1983.
- Ariav, Gad and Ginzberg, Michael J., "DSS Design: A Systemic View of Decision Support," Communications of the ACM, Vol. 28, No. 10, October 1985.
- Bentley, Trevor J. and Forkner, Irvin H. Making Information Systems Work for You, Prentice-Hall, 1983.
- Blum, B. I. "Rapid Prototyping of Information Management Systems," ACM SIGSOFT Software Engineering Notes, Vol. 7, December 1982.
- Blum, B. I., "Still More About Rapid Prototyping," ACM SIGSOFT Software Engineering Notes, Vol. 8, July 1983.
- Boar, Bernard H., Application Prototyping: A Requirements Definition Strategy for the 80's, John Wiley and Sons, Inc., New York, 1984.
- Bonczek, Robert H. Holsapple, Clyde W. and Whinston, Andrew B. "Computer-Based Support of Organizational Decision Making," Decision Sciences, April 1979.
- Brooks, Fredrick P., The Mythical Man-Month, Addison-Wesley Publishing Co., Inc., Reading, Mass., 1975.
- Brown, P. J. "Error Messages: The Neglected Area of the Man/Machine Interface," Communications of the Association for Computing Machinery, Vol. 26, No. 4, April 1983.
- Carey, T. T. and Mason, R. E. A. "Prototyping Interactive Information Systems," Communications of the ACM, Vol. 26, May 1983.
- Carlson, Eric, "An Approach for Designing Decision Support Systems," Database, Vol. 10, No 3, Winter 1979.
- Davis, Michael W. "Anatomy of Decision Support," Datamation, 15 June 1984.
- Erickson, Jonathan and Baran, Nicholas, Using R: BASE 5000, McGraw Hill, 1986.
- Freeman, P. and Wasserman, A. I., Tutorial on Software Design Techniques, 4th ed., IEEE Computer Society Press, 1984
- Frenkel, Karen A., "Toward Automating the Software-Development Cycle," Communications of the ACM, Vol. 28, No. 6, June 1985.

- Glaser, George, "Managing Projects in the Computer Industry," Computer, Vol. 17, No. 10, October 1984.
- Huff, S. L., Rivard, S., Grindlay, A., and Suttie, I. P., "An Empirical Study of Decision Support Systems," Infor, Vol. 22, No. 1, 1 February 1984.
- Jackson, M. A. and McCracken, D. D., "Life Cycle Considered Harmful," ACM SIGSOFT Software Engineering Notes, Vol. 7, April 1982.
- Jacob, Robert J. K., "Using Formal Specifications in the Design of a Human-Computer Interface," Communications of the Association for Computing Machinery, Vol. 26, No. 4, April 1983.
- Keen, Peter G. W., and Morton, Michael S. Scott, Decision Support Systems: An Organizational Perspective, Addison-Wesley Publishing Company, 1978.
- Keen, P. G. and Wagner, G. R., "DSS: An Executive Mind-Support System," Datamation, Vol. 25, No. 12, November 1979.
- Keen, Peter G. W. "Decision Support Systems: Translating Analytic Techniques into Useful Tools," Sloan Management Review, Spring 1980.
- Kent, William, "A Simple Guide to Five Normal Forms in Relational Database Theory," Communications of the ACM, Vol. 26, No. 2.
- Kroenke, David M., Database Processing, Science Research Associates, 1983
- Lusa, J.M., "Prototyping Maxi Applications on Micros," Infosystems, v. 9, September 1981.
- McNurlin, B. C. "Developing Systems by Prototyping," EDP Analyzer, v. 19, September 1981.
- Meyers, G. J., Reliable Software Through Composite Design, Van Norstrand Reinhold Company, Inc., 1975
- Morton, Michael S. Scott Management Decision Systems, Harvard University Press, 1971.
- Naumann, Justus D., and Jenkins, A. Milton, "Prototyping: The New Paradigm for Systems Development," MIS Quarterly, September 1982.
- Naylor, Thomas H. "Decision Support Systems or Whatever Happened to M.I.S.?", Interfaces, August 1982.
- Pressman, R. S. Software Engineering: A Practitioner's Approach, McGraw Hill, 1982.
- Senn, James A. Information Systems In Management, Wadsworth Publishing, 1982.
- Simpson, Alan, Understanding R:base 5000, Sybex, Inc., 1985.
- Sprague, Ralph H., Jr., and Carlson, Eric D. Building Effective Decision Support Systems, Englewood Cliffs NJ: Prentice-Hall, Inc., 1982.
- Tonies, Charles C. "Project Management Fundamentals," in Jensen and Tonies, Software Engineering, 1979.

Wasserman, A. I. and Shewmake, D. T., "Rapid Prototyping of Interactive Information Systems," ACM SIGSOFT Software Engineering Notes, v. 7, 1982.

INITIAL DISTRIBUTION LIST

|   | No. | Copies |
|---|-----|--------|
| 1. Defense Technical Information Center<br>Cameron Station<br>Alexandria, Virginia 22304-6145                                       | 2   |        |
| 2. Library, Code 0142<br>Naval Postgraduate School<br>Monterey, California 93943-5002   | 2   |        |
| 3. Commandant of the Marine Corps<br>(Code MPI-40)<br>Headquarters U.S. Marine Corps<br>Washington, D.C. 20380-0001                 | 2   |        |
| 4. Commanding Officer<br>Marine Corps Finance Center (Code MPI-LNU)<br>1500 East Bannister Road<br>Kansas City, Missouri 64197-0001 | 2   |        |
| 5. Professor Norman R. Lyons<br>Naval Postgraduate School<br>Code 54LB<br>Monterey, California 93943                                | 1   |        |
| 6. Professor Daniel R. Dolk<br>Naval Postgraduate School<br>Code 54DK<br>Monterey, California 93943                                 | 1   |        |
| 7. Commandant of the Marine Corps<br>(Code MPI-50)<br>Headquarters, U.S. Marine Corps<br>Washington, D.C. 20380-0001                | 1   |        |
| 8. Commandant of the Marine Corps<br>(Code M-1)<br>Headquarters, U.S. Marine Corps<br>Washington, D.C. 20380-0001                   | 1   |        |
| 9. Commandant of the Marine Corps<br>(Code M-2)<br>Headquarters, U.S. Marine Corps<br>Washington, D.C. 20380-0001                   | 1   |        |
| 10. Commandant of the Marine Corps<br>(Code CCI)<br>Headquarters, U.S. Marine Corps<br>Washington, D.C. 20380-0001                  | 1   |        |
| 11. Commandant of the Marine Corps<br>(Code FDD)<br>Headquarters, U.S. Marine Corps<br>Washington, D.C. 20380-0001                  | 1   |        |
| 12. Commanding Officer<br>Marine Corps Finance Center (Code M)<br>1500 East Bannister Road<br>Kansas City, Missouri 64197-0001      | 1   |        |

- |     |  |   |
|-----|--|---|
| 13. | Commanding Officer<br>Marine Corps Finance Center (Code RFAM)<br>1500 East Bannister Road<br>Kansas City, Missouri 64197-0001                  | 1 |
| 14. | Director<br>Marine Corps Central Design<br>and Programming Activity (Code FPP)<br>1500 East Bannister Road<br>Kansas City, Missouri 64197-0001 | 1 |
| 15. | Commanding Officer<br>Naval Personnel Research and<br>Development Center<br>Code 61<br>San Diego, California 92152-6800                        | 1 |
| 16. | Naval Postgraduate School<br>Computer Technology Curricular Office<br>Code 37<br>Monterey, California 93943-5000                               | 1 |





IS APP 91 36050

211316Z

Thesis

C33823 Cavallaro

c.1 An automated systems  
and project management  
system for Headquar-  
ters, U.S. Marine  
Corps.

19 MAY 90

36037 37

19 MAY 90

36037 37

211316

Thesis

C33823 Cavallaro

c.1 An automated systems  
and project management  
system for Headquar-  
ters, U.S. Marine  
Corps.



thesC33823

An automated systems and project managem



3 2768 000 65783 7  
DUDLEY KNOX LIBRARY